DOCUMENT RESUME

ED 320 550 IR 014 466

AUTHOR Herrmann, Thom, Ed.
TITLE Show & Tell. Proceedings of the Ontario Universities' Conference (1st, Guelph, Canada, May 1987).
INSTITUTION Guelph Univ. (Ontario).
PUB DATE May 87
NOTE 217p.; For the proceedings of the 1988 conference, see IR 014 467.
PUB TYPE Collected Works - Conference Proceedings (021) -- Reports - Descriptive (141)

EDRS PRICE MF01/PC09 Plus Postage.
DESCRIPTORS Authoring Aids (Programing); *Computer Assisted Instruction; *Computer Graphics; Computer Literacy; Databases; *Expert Systems; Foreign Countries; Higher Education; *Information Networks; *Information Technology; Interactive Video; Media Selection; *Teleconferencing
IDENTIFIERS Canada; *Computer Mediated Communication; Ontario

ABSTRACT

Twenty-three conference papers focus on the use of information technology in Ontario's technical colleges and universities: "The Analytic Criticism Module--Authorial Structures & Design" (P. Beam); "Computing by Design" (R. D. Brown & J. D. Milliken); "Engineers and Computers" (P. S. Chisholm, M. Iwaniw, and G. Hayward); "Designing the CAL Screen: Problems and Processes" (R. Edmonds); "A Computer Aided Instruction in Dynamics" (H. Farazdaghi); "A Computer Assisted Curriculum Planning Program" (H. Farazdaghi); "RAPPI--Communications Networking in the Classroom" (L. Hazzan); "RAPPI--Open Door" (C. Hudel); "Computer Conferencing Systems: Educational Communications Facilitators" (C. S. Hunter); "Interactive Computer Art" (M. LeBlanc); "GEO-VITAL: Computer Enhanced Teaching in Geology" (I. P. Martini and S. Sadura); "Intentional Socialization and Computer-Mediated Communication" (E. K. McCreary & J. Ord); "Issues in the Development of Intelligent Tutoring Systems" (M. McLeish & K. Langton); "Which Objectives, Which Technologies? A Decision-Making Process" (C. Nash); "Library Instructional Module" (G. Pal, L. Rourke, and H. Salmon); "Computer-Assisted Language Learning (CALL) at the University of Guelph" (D. M. Paramskas); "Computer Literacy: A Practical Approach" (L. Pearce); "The EOP: Graphics & Teleconferencing" (J. Raymond); "Horticultural Production Systems Planning and Analysis" (J. W. Riekels); "The Impact of Interactive Video Teaching" (H. Schmidt); "The Use of the Veterinary Medical Information Management System (VMIMS) Database in Teaching" (T. Stirtzinger); "A Computer Science Summer Camp" (D. A. Swayne, J. VanDuren, and M. M. Shepherd); and "A Prototype Expert System for Teaching Brain/Behaviour Relationships" (M. Szwarc, S. Bandyopadhyay, G. Lasker, and D. Reynolds). References are provided in each of the papers. (GL)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

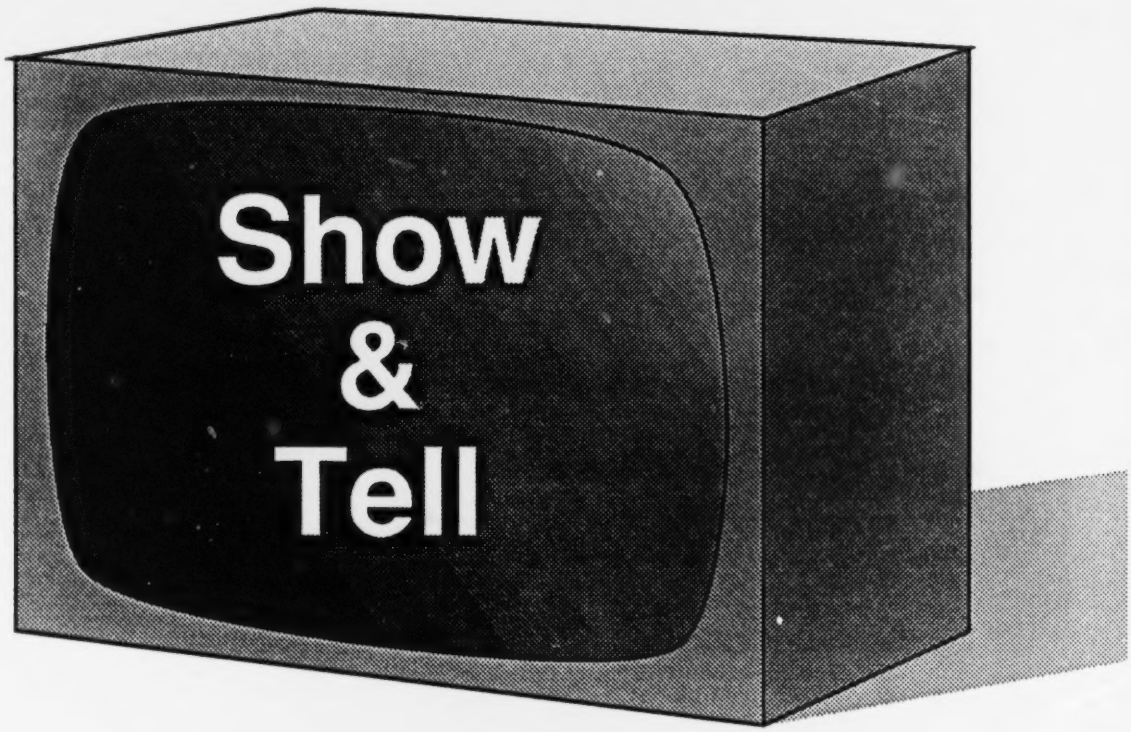
U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

X This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

ED320550



BEST COPY AVAILABLE

papers from the conference

May 1987

UNIVERSITY of GUELPH

IR014466

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
Ian K. Easterbrook

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

Preface

The 1st Ontario Universities's SHOW and TELL was an attempt, held at the University of Guelph during May 1987, to bring together those faculty using Information technology to augment their courses and both tell as well as show others what worked and what didn't.

The conference consisted of a series of 4 or 5 ten minute presentations followed by an equal period during which individuals were able to demonstrate their applications. In this manner a general description was complemented by a poster type demonstration for those interested in a particular area or application.

The present volume is an attempt to move these rather informal presentations into a generally more reflective and scholarly format yet maintain the spirited and useful information exchange that the SHOW and TELL has so far engendered.

I wish to thank both the authors and the staff of the Information Technology Support Group of the Office of Educational Practice at the University of Guelph for their effort in compiling this volume. They, the authors and staff, whose continued interest and support of the free exchange of information are leading to the effective and efficient application of Information Technology in the university setting. They deserve the credit for any success the Show and Tell or this volume might enjoy.

Thom Herrmann
Coordinator

TABLE OF CONTENTS

The Analytic Criticism Module (ACM) - Authorial Structures & Design 1
Paul Beam

Computing by Design 15
Robert D. Brown and John D. Milliken

Engineers and Computers 21
P.S. Chisholm, M. Iwaniw and G. Hayward

Designing the CAL Screen: Problems and Processes 31
Robert Edmonds

A Computer Aided Instruction in Dynamics 43
Hadi Farazdaghi

A Computer Assisted Curriculum Planning Program 47
Hadi Farazdaghi

RAPPI - Communications Networking in the Classroom 53
Linda Hazzan

RAPPI - Open Door 63
Chris Hudel

Computer Conferencing Systems: Educational Communications Facilitators..... 71
C. Stuart Hunter

Interactive Computer Art 77
Michael LeBlanc

GEO-VITAL: Computer Enhanced Teaching in Geology 91
I. Peter Martini and Steve Sadura

Intentional Socialization and Computer-Mediated Communication 101
Elaine K. McCreary and Jonn Ord

Issues in the Development of Intelligent Tutoring Systems 111
Mary McLeish and Karl Langton

Which Objectives, Which Technologies? A Decision-Making Process 127
Chris Nash

Library Instructional Module 135
Gabriel Pal, Lorna Rourke and Helen Salmon

Computer-Assisted Language
Learning (CALL) at the University of Guelph 141
D.M. Paramskas

Computer Literacy: A Practical Approach 157
Laurie Pearce

The EOP: Graphics & Teleconferencing 165
Jacques Raymond

Horticultural Production Systems Planning and Analysis 177
Jerald W. Riekels

The Impact of Interactive Video Teaching..... 185
Herminio Schmidt

The Use of The Veterinary Medical Information
Management System (VMIMS) Database in Teaching 193
T. Stirtzinger

A Computer Science Summer Camp 199
D.A. Swayne, J. VanDuren and M.M. Shepherd

A Prototype Expert System for
Teaching Brain/Behaviour Relationships 211
Mark Szwarc, Subir Bandyopadhyay,
George Lasker and David Reynolds

THE ANALYTIC CRITICISM MODULE (ACM) - AUTHORIAL STRUCTURES & DESIGN

This module, the ACM, has been produced under the auspices of the Computers-in-Education Committee of the Ministry of Education and has been written in C to operate on the UNISYS ICON. It requires both colour facilities and the Window Manager routine to function.

THE ACM STRUCTURE IN THE CONTEXT OF ICON SYSTEMS & NETWORKING:

The ACM's purpose is to permit teachers, particularly in language studies, to create very detailed, interactive lessons in computer-aided learning through a series of screen displays, menus and system commands, without being required to do any programming or database construction within the module. We believe that this is a major advancement in making teacher-specific CAL available in a congenial and comprehensible format. The tool described in this paper is in prototype and we commence testing it in schools in September of 1987. The guiding pedagogy has been to place a subject expert behind the left shoulder of a student in the examination of a short text or document. The passage may be of the instructor's choosing and may be of any length, though all our experience has been that brief texts, because of the depth in which they can be examined and explicated, be undertaken initially.

Because of the pre-eminence of the text in this learning model, it is virtually impossible to not have some or all of it on the screen, in a central location, at any given point. The student (or instructor in the lesson structure) is constantly indicating some aspect of the passage and making a comment or query about it. The teacher does so by a series of pre-defined questions and suggestions which make up the module's options and by responses to student choices or questions. The student makes selections from the passage, choices among topics and definitions and adds comments to a file which the instructor reads subsequently and from which he

or she can respond to the student, incorporate answers for an entire class and make modifications to the lesson, where such are warranted.

The creation module is the first of four, linked structures. In it the teacher prepares a comprehensive explication of any facets of a text on which comment is deemed necessary. It, in turn, generates a user module which resembles, as closely as possible, the appearance and sequences of the creation unit, to enable both instructor and user to look at the same formats by the same commands. There is an obvious advantage for both parties in close resemblances between what the teacher makes and what students actually see in the resulting lesson. Behind this premise lies our hope that students will be drawn in as aids and assistants in the creation process and in amplification and expansion of all aspects of a given lesson; where they are familiar with the format and the patterns of commands and devices, they will be the more practically prepared for this educational step, the more the creation design approximates the lesson structures on which they have been trained. A third component of component of the ACM, seldom seen by users, is the "datafile", the structure of teacher lessons, comments, choices and questions, as well as the large libraries of theory and help which can be linked to individual aspects of the lesson. These can be involved, modified, expanded and re-ordered by the instructor. The ACM runs under the Ministry's AMBIANCE program which establishes teacher and student accounts within a class and operates the installation, availability modifications to lessons within a given school's ICON system. The teacher is able to specify conditions within the ACM module to enable the user to collect information - pieces of text, definitions, patterns which develop within the passage - and to store these in personal accounts within the "AMBIANCE" system for future use in assignments, essays or study. This third structure, AMBIANCE, is the system version the class and the instructor may send messages, examples of student work, progress reports, announcements or lesson

instructions to all users through it. Reciprocally, groups of students may be organized through it in turn to carry out modifications and new developments on the lesson itself.

The largest structure of the four is "edan", the Ministry's computer communications network, linking 100 Ontario schools and research units via phone lines and datapac throughout the province. Within it the RBCS system permits teachers (and students) to store, create, exchange, and modify information, and CAL materials in particular. As this network develops and expands, it will be increasingly possible to exchange large, compiled lessons among institutions. At this level in the available technology, CAL of the kind described herein becomes increasingly efficient for two reasons. In the obvious case, the time and effort required to prepare lessons of the detail required by the ACM can be rewarded by the reciprocity of useful materials from other teachers, in other schools; once this process has begun, along with entire lessons being available in any school with ICON facilities, teachers will be able to utilize "parts" of lessons - the kinds of internal CAL mini-modules with applications to a number of lessons in a particular subject area. For instance, an inter-active module which explains "metrics", provides examples by user choice and then permits the student to try to correctly identify stresses and meter from a model which is then assessed and responded to by the system, may have application in a lesson on metrics, as a subjunct to the teaching of a Shakespearean sonnet and to work in linguistics. Once created by an instructor or user-group, this small lesson can be incorporated in larger modules by being copied from a "library" of language materials made available to any instructor by computer networking or by the simple process of copying it onto a floppy disk. By this process CAL takes on a richness of depth and resources not achievable by an individual teacher's effort, no matter the degree of that person's commitment. All CAL lessons in this format can be easily copied and can be modified by the ACM

in any location to adapt them to the particular needs of an instructor and a course.

THE INSTRUCTOR MODULE OF THE ACM:

Because the early versions of the ACM will be unfamiliar to teachers in both the theory and terminology of critical analysis and in the processes and techniques of operating the module, we have attempted to develop the HELP functions in detail and in some syncronicity to each other. They are easily accessible from any location in the program by the use of an appropriate icon on each screen. To test the creation unit's functions and to provide an initial model from which novice instructors can commence work, we have developed a detailed analysis of Thomas Hardy's "Afterwards", a twenty-line piece of verse. New users are encouraged to consult it for both models of how to approach a passage - what questions and information to use, as well as what topics and techniques to employ and for some insight into the critical theory behind parts of the module's design. No one is obliged to use any of the theory or specific literary topics provided or use any order of presentation other than one congenial to each developer's insights and emphases but the lesson shows a number of subjects and methods and at least suggests some ways of approaching a text.

We have begun development of a library of technical and literary terms which will be available to all users as a kind of "lexicon" from which they may draw materials for inclusion in a particular lesson or to which they may refer their student users for more extensive reading. Definitions within it - as those within an on-line dictionary in the near future - may be modified as they are brought into specific lessson modules to render them more appropriate to particular contexts. An author may omit tangential references, add germain examples or comments and link the inlesson definitions to inter-active "sublessons". For example, an instructor

might include a sonnet as an optional example in the literary definition of the term, make references to other sources as bibliography and then give the user the option to go to a selftest of the concept of a sonnet in which meter, line length and rhyme scheme could be assessed by the user's correct identification of each characteristic, all from a small program within the lesson itself. Such an extensive definition can be easily transferred from the lesson in which it had been developed to the library and thence to any other designer who might have use for it in any lesson. We hope in this way to assist new instructors in the future and, because the development process thoroughly executed can be an extensive one, to reduce repetitious aspects of it - the re-entry of common definitions, for example - by having them easily available for all users. Larger lessons can be exchanged by instructors' agreements and this will render development time even more effective as users in a number of locations come to benefit from a variety of analyses on different subjects shared in common among their creators. Teachers will then see large benefits for their individual efforts by such exchanges and students will have a variety of materials from which to learn and, we hope early in the process, to which to contribute.

Computer-aided lessons can be very specific, focussing on one or more subjects -- meter, for instance. They can also incorporate a wide range of critical devices and perspectives and, obviously, they become increasingly complex as comprehensiveness is attempted. One of our major recommendations is that they should be started on a modest base and then be built in light of experience, student reactions and perceived needs. In this sense, optimal efficiency will be achieved by teachers using texts with which they are familiar and which they have taught before. To discover what possible answers come to users' minds, one needs to mark several sets of papers on a given topic. Within the ranges of responses it becomes clear that two or three perspectives or opinions occur frequently and these

can be anticipated and responded to in advance so it is possible to give both directions and answers to most users as well as to provide directions to new topics as extensions of what they express in their answers. Of course not all questions and responses can be anticipated so users must be assured at an early stage that their responses can be dealt with by the instructor at a point in the near future. This is accomplished via the comment facility from which the user can communicate with the instructor and, under certain circumstances, other people working within the module. A function key invokes the editor, the user comments, at whatever length, and the resulting material, with a marker of its location within the topics and screens of the lesson, is sent to the instructor account for appropriate action. The long-term good effects of this process are threefold: the lesson is improved over time as user performance and comments point to weaknesses and strengths to be modified or enhanced. As well, users participate in the learning process and, depending on the degree of use of the device, they create an environment more of a tutorial than a lesson in isolation or a dialogue. Many student comments prove to be useful responses in their own right and can be incorporated in the larger lesson as optional perspectives or as fully integrated parts of the ranges of answers to other users' questions and performance. It is my strong hope that students will be drawn in time to be full participants in the creation aspects of the lesson, coming to develop many of the detailed support programs and facets of the main work. For instance, a student who comes to see grammatical links among patterns within a text can go on to develop these in full detail from an instructor's limited range of examples; further extensions can incorporate inter-active questions from a user about other grammar-related concerns and larger structures of grammar of general use to a number of modules on language. These in turn can be transferred, in whole or in part, to be modified and adapted into other lessons, on the same or on remote CAL systems.

It has been my experience that students, given definable projects from topics of particular interest to them, benefit from the atmosphere of personally directed learning in the material itself; they must conceptualize a topic, limit its dimensions, get and evaluate material and express it in a comprehensive form to appeal to many users. They learn about both the format of expression - the integration of ideas into spaces and sequences - and the system on which the work is displayed. The effect of performing real instruction seems to be a tonic to motivation, particularly as we have been able to have students explain what they are attempting to other students.

USES FOR THE ACM: Ranges of subject matter.

Initially we had conceived of preparing a single, very complete module to teach users the major categories and types within analytic criticism, using as our only text Thomas Hardy's Afterwards, a poem of some twenty lines in five sentence-stanzas. As we came to devise the necessary editors, messaging systems, libraries and movement and checking tools, it became apparent that, with a prodigious effort, we could turn over to users a very flexible and expansive CAL creation system. In effect, what we had to provide for ourselves as authors could be packaged as a set of creation programs to permit any instructor to perform similar operations on any text. If this were so, then, presumably, any instructor, using any text, could perform similar operations on any subject. Auto mechanics, with a strong integration of graphics functions, would be as achievable in this search and question format as analytic criticism of literary texts in English. We have moved most thoroughly on our subject matter, of course, but we have also become intrigued with other potential uses and plan to explore several, particularly in the teaching of business-related materials in the near future. In the complete expression of the ACM, as we conceive it, an interested instructor will be provided with the following:

- the ability to turn over to students within the ICON/Ambience environment a stand-alone facility for learning analytic criticism. This will include the module of Afterwards, as a fairly complex exposition of critical theory, a tutorial session which takes a novice instructor across the major critical points and system devices through a short passage entitled TO A FISH OF THE BROOK.

- Complete lexicon files for its vocabulary, grammar and semantic expressions for all words in the passage.

- A set of some twenty topics among which the user may browse, with appropriate references, based on user performance, to other, related topics and a set of inter-active sub-lessons from which the user can assess his or her understanding of the topic under consideration.

This is completed by an extensive series of help files which define and discuss the various topics which, together, make up the theory and application of analytic criticism.

Using just this module, the teacher may -

- turn students immediately to the practice of criticism by merely requiring them to work through the module on Afterwards by themselves, using only the questions and aids of the module.

- work through the module with the students as a class project by a combination of conventional and CAL lessons.

- require the students to complete their analyses from either mode by a formal essay on the features of the passage.

- commence work on a class or instructor lesson on another passage, using the theory/topic links provided by the system by invoking the lesson structure provided as a template within the ACM. Here the instructor may select from among the fifty passages available within the ACM or may enter a text of personal choice against which all the options of the lesson creation unit may then be employed.

- use the ACM process to either modify the critical theory files individually to reflect the instructor's personal view of Afterwards or select among them to incorporate desirable points and areas of the theory into the instructor's lesson on a new passage.

- employ the structure of the ACM to create entirely new theory on Afterwards or any other passage of the instructor's choosing.

- apply the ACM structure to an entirely different subject, using the topic function to create an entire lesson on the operation of a turret lathe or alternate strategies for re-fighting the Battle of Waterloo.

In all these conditions instructors may invoke existing libraries of terms, definitions, information trees and devices as they exist in the database or as they are created by individuals or classes to respond to specific situations. Existing bases can be modified, expanded and adapted to new subjects so that a lexicon of Middle English terms created for a module on the Canterbury Tales may be copied and incorporated to run in a lesson on Medieval law, with appropriate additions for that subject matter.

A SURVEY OF OPERATIONS

The MENU screen offers the following choices:

1) Create Lesson (Teacher Module)
2) List Existing Lessons
3) Critical Technical Overviews
4) Run Lesson (Student Module)
5) Selections Of Text
6) Quit

Of these the most important is the Lesson Creation option. Presuming that the instructor has found in the list of available passages a text of preference (the alternative is to enter a selection from the editor), the creation unit is now prepared to begin computer-aided learning materials preparation. For beginning instructors, the process need not be complex. Presuming that the text is one to be studied in conventional classroom discussions, the initial questions may well be those the teacher would use to open the subject. Here, by the selection of topic, the instructor may choose to prepare material within one of the areas of analytic theory structured by the ACM. SETTING, for instance, would include ancillary files which are automatically linked by the Help and Define icons to explanatory materials available to the user throughout both that topic and the lesson. Should the instructor wish to modify these or to create a topic not included among those of analytic theory, the theory files must be accessed and modified or else be replaced with appropriate new materials of the instructor's preference. Where no such materials are created, no structure is invoked by the system to link the user to assistance beyond that of the module's operational Help feature itself. Oncethe topic has been selected or devised, the instructor enters question mode, asking as many questions as seem appropriate to the topic, leading the user by these to an understanding of the topic and its relevance to the text - much in the pattern of

lessons with which the teacher is familiar. After the question has been entered, the instructor is prompted to select those elements of text which answer it. These may be selected in any order, as words or phrases, and may be modified at any point in the process; however, once the series of right/wrong answers has been set, the question is established and can be changed only by a new question being designed to replace it in its entirety. With the selection of each right or wrong term, the instructor may make a statement about why that term is correct (or not). This statement is read by the user only after the question has been attempted and the answer has been displayed within the text itself, at which point the user may select any parts of the answer - right or wrong - to receive the instructor's explanation of why that selection is appropriate or not. The student may check all significant terms in the passage in this way, commenting to the instructor at the end of each to indicate agreement or to continue the discussion. Student choices can be surveyed to look for patterns useful in the re-defining of the lesson and the honing of the questions.

In the topic and question creation workspace the instructor can be as thorough as time, subject and inclination permit. Audience sophistication, the uses to which the lesson will be put and the individual's sense of the need for comprehensiveness determine just how much effort is required to begin profitable use of the module by students. We anticipate that instructors will require time and experience to develop effective techniques for interrogation and commentary so we offer models, examples and advice at both the topic and question levels. The final stage of preparation is the comment prepared by the instructor to conclude the series of questions on a particular topic - this is the summation of the directed argument made by the patterns of questions and answers. Here the instructor brings together the cumulative points and relates them to both the theme of the passage and other topics. The structure of the ACM has been kept intentionally non-hierarchical. This

places a consequent responsibility upon the instructor to add as much sequencing and form as are requisite for the topic. Here too, we feel that instructors will achieve their best personal levels of teaching from a minimum of organization imposed upon them by the system itself for non-pedagogic reasons.

Tracking or the surveillance of student performance, has been minimized, though we may add more extensive records keeping in later versions. At present students receive, along with the on-screen display of correct-wrong-missed, a numerical list in the lower right corner of the text display at the point in the lesson where the question responses are assessed by the system: 9/ 3/ 5. Instructorsmay set quantitative or qualitative ranges of user performance at which to respond with appropriate encouragement and directions as a user receives the system response to a question. Because the student has thought closely in developing an answer, these may be the most effective locations to make relational and thematic points. Here too, the emphasis is on the instructor's decisions and abilities to direct, rather than the system's rigidity of structure to enforce. We feel a single sequence or order of process may not be necessary or appropriate to the material or the natural learning sequence for all users. The final level of instructor's choice is the ORDER LESSON option which permits the instructor to re-arrange the sequences of both the questions and the commentaries which accompany them. Here too the materials can be modified and erased in response to both student patterns of performance and subsequent inspiration on the part of the instructor. One enters this area from the Lesson line and movement is made by menu selection of the appropriate files. This area can be accessed only from the instructor mode, obviously.

LIBRARIES

Many of the topics in analytic criticism have elements of language which will

apply to virtually every text selection - common definitions within grammar, literary convention, historical linguistics, to name a few. Here we have developed the concept of a library by creating what are, in effect, small lexicons specific to each subject. Initially these may be definitions of grammatical terms, for instance. The instructor accesses these and selects the appropriate materials to include in the in-text definition of a word. In this way instructors' efforts are saved and lessons reveal a similarity in their background materials, though each author may modify any term from a library and, in fact, may add elements to any library. In the future these may come to include graphics, movement and even sound; for the moment, they can include interactive examples and user self-tests in concepts like "meter" for instance. They are added to the module via the DEFINE icon in any appropriate "topic".

IN CONCLUSION

How innovative instructors may become will depend on their perceptions of what they are doing, as well as on the flexibility and clarity of the module itself. Our efforts have been directed simultaneously to its design, features, relationships and the creation of materials to serve as examples and models. I strongly suspect that we have been oblivious to many opportunities for inovative expression as we have concentrated on design of the active parts for instructor use. Mindful of this, we are most anxious to have the ACM in beta test for final distribution in a teaching environment in early September so instructors and students can validate its educational uses.

COMPUTING BY DESIGN

Robert D. Brown and John D. Milliken

School of Landscape Architecture

University of Guelph, Ontario

This is a record of observations about the use of computers in the teaching programs of the School of Landscape Architecture, University of Guelph. In many ways it is representative of experiences elsewhere in instruction programs in the design field. In comparison to programs of a more scientific orientation, for example, landscape architecture was slow in integrating computing into day-to-day instruction. A number of reasons for this may be cited. Among these:

(1) Early computing activities focussed heavily on 'number crunching'. While landscape architects do deal with statistics and some civil engineering, these activities are not central to design. Therefore, no pressing need was felt by either faculty or students to 'plunge into the electronic age' at a time when many others did so.

(2) The area of landscape planning involved an early thrust into exploration of computing in landscape architecture as mapping and geographic information systems became available. The Harvard 'Symap' program was, perhaps, the first use of computing seen to be central to the work of some landscape architects. This, though, involved only those instructors and students with a special interest in such work and the main stream of the profession (and the School) continued to rely solely on traditional teaching/learning techniques.

(3) While early costs for computing were small and heavily subsidized by the University, budgets at the School level were increasingly restrictive and priorities were placed elsewhere by a faculty who were generally unfamiliar with and intimidated by the 'main frame' and 'FORTRAN' and 'WATFOR' and the jargon which had grown up around the intense computer users.

(4) Hand-drawn images have been the primary design tool for centuries. There appears to be some subtle connection between the mind, hand and drawing of the designer that the computer tends to interfere with, despite the very powerful and useful Computer Aided Design (CAD) systems now available. Designers gain much manipulation capability with these systems but lose the open-ended freedom of exploration afforded by a soft pencil and a large sheet of paper.

Recent advances in hardware and particularly software have made computers much more available and desirable for designers than ever before. Economics also comes into play - as soon as one

office has a computer system that allows practitioners to produce drawings more quickly and inexpensively, then the pressure is on others to become computerized as well. Our students now realize that they must possess computer skills when they graduate with a landscape architecture degree if they are to be employable in the profession.

We still have problems, however, with students' aversion to numbers, their perception that computers are rigid, stupid machines, and their concern about the loss of hand drawn graphics (which tends to be a large part of the magic in many students' designs).

This sets up a contradiction between what the students know that they need to learn, and what they would rather learn. Therefore we are faced with somehow introducing students to computer use when, as yet, most of them would rather put their energies into learning the "magic" of hand-drawn graphics.

We start with a deliberate ploy. Our site engineering course is used to introduce students to computers for the first time. Students tend to have an aversion to this course anyway because it requires them to use numbers. But the course is made much more palatable by introducing students to very friendly computer programs which do the several types of engineering calculations needed. Students are 'tricked' into using the computer as a calculator without worrying about the fact that it is a computer.

Everybody gains from this. The students learn more site engineering because they can practice solving each problem several times to make sure they have the best solution. These are problems with no 'right' answer, but 'better' and 'best' answers and the speed of the computer allows them to experiment with input variables and to test the results over and over again. Students used to solve a problem once, and hand it in (right or wrong) due to the amount of time invested in calculations. Now they explore many solutions and select the most appropriate one. Much more practice is gained and incidentally, students learn how to turn on the machines and use them in a rudimentary way, without the usual nervousness or fear of "the computer".

Once the students have come to accept computers as friendly calculators we then ease them into computer graphics. We have paint programs that are very easy to turn on and use, and students only have to learn one new skill - how to move the mouse around and make the pencil on the screen draw. It is a lot of fun for them. We treat it like a game and they take turns drawing 'things' on the screen. They laugh at each other's lack of skill and quickly gain a very positive attitude towards computer graphics. They still don't need to know anything about computers or even printers. They simply photograph their pictures off the screen, a skill that they are already comfortable with and enjoy.

Soon they are creating text slides for presentations, quick graphics, or just trying things out in their spare time, exploring abstract ideas and the computer's capabilities.

Students usually become aware at this point that word processing can make their life easier. The teaching program requires a lot of group work and usually one student in the group knows how to type and wants to try out, or has already learned, word processing. Soon the whole group is in the computer room working on computers, but thinking 'very sophisticated typewriter'. They begin to see the power of the computer to cut and paste. Our students are notorious for cutting and pasting by hand until their report is just right. Again they use the computer as a tool and they don't see it as an intimidating machine.

From the faculty standpoint this is all very positive. Reports are clean, with no erasures, no spelling mistakes, are well formatted and on time... a delight to read and evaluate. Students are producing the best work possible without the time and other limitations of having someone else type it for them.

As always, though, software that is very good for one thing is generally not very good for others. Students soon see that they cannot use the paint programs, for example, to generate good 'hard-line' graphics. This is when they are ready to move on to something less 'friendly' but more useful in their education.

They are now quite comfortable, through 'osmotic learning', with floppy disks, saving files and other operations, all learned in a non-threatening manner. We then introduce them to the most powerful microcomputer software available to landscape architects - CADD. We tell them these programs are less friendly, but much more flexible, that they can do virtually anything they want given sufficient practice time and knowledge. We have found that students can expect to spend about 40 hours of computer time with CADD to get really comfortable with it and then off they go. We help them get started, keep the manuals nearby and give them a project which gradually introduces them to the many drafting tools available in the software.

After this point, they are very comfortable with computer use. They are learning much of the jargon, mainly through simply hearing it used by others, and are beginning to help the newer students get started. This is an important process, almost a tradition in other activities in the School and it is beginning to happen with computing. Many of the technical skill that students learn in the school have always been learned from other students, (quick and dirty graphic techniques, how to cut and paste, how to fix up a drawing when you have accidentally sprayed it with black paint instead of fixative or spilled coffee on a corner and so on). Now information on computers is beginning to be passed along in much the same way.

There is great potential for computer use in landscape architectural design education. The advantages clearly outweigh the disadvantages.

Problems still exist, however, including:
(1) how to encourage other faculty, particularly ones who are older, and more established in their ways, to use the computer in everyday teaching, and

(2) how to fund sufficient numbers of computers to allow access by all students.

Some of the disadvantages of computer use are:
(1) Students sometimes run computer procedures because they are available and not because they require them which leads to what we call 'analysis paralysis'.

(2) Computers tend to be inflexible, going against the nature of a designer who tends to want much greater flexibility in design tools.

(3) The start-up time to get design students involved is very high, with some Computer-Aided Design and Drafting (CADD) programs requiring a minimum of 40 hours of familiarization time.

Computer use offers the students many advantages in their design education. Some of them are:
(1) They can explore many ideas very quickly.

(2) They are not 'stuck' with a solution because of a large amount of time committed to it.

(3) They can explore the implications of a change in their design without having to redraw everything by hand.

(4) The computer allows interaction whereby the student can iterate to resolve a problem most effectively.

(5) The quality of the finished product, is often increased through computer-generated output.

(6) The computer allows students to extract layers from their designs and look at patterns in isolation. For example they can evaluate the pattern of just the roadways, or just the trees, or just the drainage.

AT bottom, we have found that the best way to make designers computer literate is to 'ease' them into computer use without them having to ever think 'computer' or about FORTRAN, or MAINFRAMES, or even about very many numbers. They learn to think of it as a fast calculator, a smart typewriter, a colorful image maker and finally a very handy design and graphics tool.

ENGINEERS AND COMPUTERS

P.S. Chisholm P.Eng., M. Iwaniw P.Eng., G. Hayward P.Eng

The examination of computers and their applications can take place in the general context of information technology; alternatively, there is an on-going examination of computers and their applications that relates to specific fields of industrial practice such as architecture, communications, engineering, library science and medicine.

Our article refers to contemporary engineering practice. Our objective is to propose that teaching/learning about information technology can lead to personal learning outcomes that are more durable than the technology itself. Our proposal incorporates four relationships between engineers and computers that occur in contemporary engineering practice, and which are cast in the form of a general learning objective about information technology for undergraduate engineering students. Student experience with the objective is described in terms of curriculum content, and some examples of computer applications in engineering practice are given.

The content is intended to show that the learning outcomes wanted are personal in nature. Further, personal ability at analysis, application, synthesis and optimization, among others, is separate from the medium of information technology that contributes to its learning. Finally, in operational terms, the abilities have the general form of information processing that can be applied independently to new information as it is encountered.

1. Four Relationships

The first, and most pervasive, relationship between engineers and computers is due to computation content in engineering practice. The microcomputer is the current engineering computational aid, having replaced the programmable calculator and its antecedents, the slide-rule and logarithm table.

Next, the computer helps the individual at learning how to learn. This abstract relationship entails experiments with translation, interpretation, and extrapolation of information to discover(learn) which principles are required to identify(learn) relationship between variables or perhaps recognize(learn) trends in data that are otherwise not immediately evident.

The third relationship is set in the context of mathematical modelling (imitation) of real engineering systems. Modelling is rich in learning content, as it helps the designer to learn about the necessary relationships between the parts of a system which enable the system to operate in the way intended. Modelling is a simulation of system operation, and it is essential in learning the admissible ranges of system element parameters for design of real systems.

The fourth relationship is determined by the use of computers as parts of engineering systems. As components in control systems, computers continuously monitor the operation of aircraft, breweries, petroleum refineries, traffic control systems and many other familiar examples. Beyond their immediate control function, they help the designer to learn about the operation of real systems and contribute to new information required for learning how to improve upon existing designs.

Instruction about the preceding relationships is organized according to the following general learning objective.

2. General Learning Objective

Learning about the four relationships between engineers and computers can be assessed by observation of student performance in response to relevant assignments. More specifically, observable student behaviour together with objective results or products from student behaviour, provide a basis for evaluation of learning. Thus the following general learning objective is stated in behavioral terms and comprises five parts.

a)Who is required to perform?

Teaching/learning about engineers and computers is continued through semesters 1 to 8 inclusive. All undergraduate engineering students are continuously encouraged to develop independent learning ability within the curriculum structure described in Section 3.

b)What performance is required?

All students are required to independently program the solution to proolems assigned in different subjects (statics, design, economics, etc.) using a high level computer language. They are also required to run the programs and evaluate the results. This task incorporates both original programming by the student and the use of distributed or "off-the-shelf" software.

c)What constitutes an acceptable response?

Criteria for evaluation of student performance differ among courses. In general, evaluation is directed at measurement of independability in the following:

- verbal and written (printer) communication;
- analysis and synthesis;

- modelling of engineering systems;
- production of new knowledge from information processing;
- evaluation of personal work; and
- use of computers as components of engineering systems.

d)Curriculum and constraints?

Curriculum design provides a progression from formative through summative learning about relationships between engineers and computers. This progression is described in Section 3 by relating learning outcomes to different points in the curriculum.

e)What facilities are required for the student to perform?

Facilities provided to assist learning are described in Section 4.

3. Learning Sequence and Curriculum

Our purpose in the School of Engineering is to train engineers in the use of the tools that he/she will need to fulfil his/her job - design. Traditionally, these tools have been the T-square, or a transit, or a handbook. Today, the computer is one of the most important tools that an engineer uses. Upon graduating, the student should be both familiar and comfortable with a mainframe and micro-computing environment.

The Engineering program runs for four years. The first year provides an introduction to computing; the second year provides applications. Year Three goes into analysis, and introduces design and synthesis/optimization, while Year Four provides practical experience with design and synthesis at the summative level of learning.

a) YEAR 1

The first year program includes an introduction to the computing environment at the School of Engineering. The micro-computer lab, which is

available to all engineering students, is equipped with 15 IBM PC's, and 20 carrels with power supplies for portable computers. Peripherals include printers, plotters, and modems, all networked through their serial ports. There are also two terminals to the mainframe installation on campus.

By the end of Semester One, the student has been required to use the word-processing, BASIC programming, and plotting capabilities in this lab. As well, they have been supplied with TCoSy accounts to introduce them to computer conferencing, and special accounts on the mainframe WATFIV utility. By the end of Semester Two, the student is expected to have written and run simple BASIC programmes, and to have used off-the-shelf software in this environment.

b) YEAR 2

During the second year, the student learns the three major applications of the micro-computer within the engineering discipline: acquisition and control, computation, and presentation. The data acquisition and control function of the computer is used in the materials science course, where the laboratory equipment is in the process of being instrumented. Data will be acquired, manipulated, and displayed in the lab during the course of the experiment. The "number-crunching" abilities of the computer are put to use in the numerical methods course, where the student develops a library of his/her own programmes, which can be used subsequently whenever numerical analysis is required. The presentation aspect is addressed in the design course, which requires professional quality reports, and the implicit use of the computer's word-processing capabilities. By the end of Semester Four, the student should be familiar with and have personal experience with these three applications of the computer.

c) YEAR 3

Year Three deals with analysis, and the student learns how the computer can aid in engineering design. This process involves four steps:

1. presentation of the theory of some physical phenomenon;
2. illustration of the theory using "manual" examples;
3. presentation of a computer model or simulation of the phenomenon, which is verified by repeating the examples of Step 2 on the computer; and
4. the accepted use of the computer model.

It is also in Year Three that the student learns to regard the computer as hardware; i.e., as another component of the system that the engineer is designing or using. In the micro-processor course, the student is supplied with a micro-processor, and learns to assemble a control circuit using that processor and to programme it in machine code to perform a particular task.

At the end of Semester Six, the student should regard the computer, whether in its software or hardware aspect, as just another engineering tool, which can be incorporated into the design process.

d) YEAR 4

Year Four involves synthesis and design, and the student is expected to design a system using the tools and scientific principles learned previously. The use of off-the-shelf software is accepted as a matter of course. The entire process culminates in Systems Design, where the student does data acquisition, numerical analysis, word processing, and produces display documents for presentation. As well, the projects selected usually involve the hardware aspect of the computer with a microprocessor

incorporated into the design itself.

By the end of Semester Eight, the student is expected to have a complete acceptance of the computer as simply another engineering tool with which to accomplish the simulation, control, and presentation of design.

4) Computers as Parts of Engineering Systems

a) Computer Courses

In addition to using computers for engineering computations and simulation, students may incorporate computers into the systems they design. In measuring instruments, non-linearities and noise may be removed by computation. In control systems, the computer senses upsets to the process, calculates the appropriate corrective action and carries out that action. This latter operation imposes more stringent constraints on the computer than those imposed by data processing tasks. The most significant constraint is the computer response time - disturbances must be controlled as they occur; if the response is not fast enough, the system may fail catastrophically. Two courses in the engineering curriculum at the University of Guelph address these applications of computers specifically.

In their third year, students take a course in microcomputer fundamentals. In this course, the inner workings of computers are examined and interface circuits to sensors are designed. The students are required to build the interface circuits in the laboratory and to write the software required to operate them. Each laboratory work station consists of a Radio Shack Colour Computer with 16K bytes of memory. A custom-made interface board is installed in each computer and is connected to the students' circuits which are built on a plug-in prototyping board. A Radio Shack Editor/Assembler cartridge is installed on each custom board to provide a

good programming environment for the students. This computer system is less expensive and easier for the students to use than most of the single board computers.

The software for the individual projects is written by the students in Assembly language. Although this language is rather complex, it gives the user complete control of the computer. The students gain some appreciation for the capabilities of small computers and are exposed to a different type of language.

In their final year, engineering students may take an elective design course which examines advanced control theory and digital control systems. The implementation of controllers based on this more advanced theory may be feasible only if computers are used. This is particularly true when the controller must adjust several parameters simultaneously. Computer control systems are also very versatile: only the software need be changed to completely alter the type of controller implemented.

This versatility is well-used in the laboratory part of the course. The students are required to write a control program in PASCAL and test it on a simulated process. Then they must modify the program to implement classical PID controllers as well as controllers designed by Z-transform techniques. Each work station consists of a Radio Shack Colour Computer with 64K bytes of memory and two floppy disc drives. This computer was chosen because a multitasking operating system, OS-9, was available. An analog interface (Byte Craft Limited, Waterloo, Ont.) is connected to the RS-232 serial port. This interface may be used with any computer with a standard serial port.

b) Teaching Equipment

One development project being carried out by the School of Engineering

is an instrument system for the Materials Science and Soil Mechanics laboratories. Here, experimental data is collected by a central computer and relayed to display stations at each of the experiments. This allows the students to see plots of the data develop as the experiment progresses.

The central computer is an Amiga A-1000 with a Daytronics 9010 data acquisition system. The display stations are Radio Shack Colour Computers. The project requires the development of a local area network to transfer commands and data between the central computer and the display stations. The hardware has been built for approximately $10 per station. The software, however, is expensive. A considerable amount of time has been spent on the project and more is required. One rule of thumb states that software costs about $65 per line of documented, debugged code. This is not far from the truth.

5. Summary

The four relationships between engineers and computers are concrete and provide a structure for learning and using information technology. Together, these relationships present an interface between the person and the computer system which is rich in opportunities for learning at the enhanced rates and precision of electronic information processing which is not achievable by human processing. More generally, the interface provides a learning environment for the practice of personal abilities that is not available elsewhere. Further enhancement of the learning environment will result as teaching methods change and new systems are developed.

Designing the CAL Screen:

Problems and Processes

Robert Edmonds
Sheridan College

Abstract

As real-life students meet real-life Computer Assisted Learning, the importance of quality, interpretable graphics becomes more and more apparent. Although they work in an essentially visual medium, most CAL developers do not have professional visual design training. As a result, solutions to instructional design problems seldom take full advantage of the delivery medium's visual nature.

In order to realize this neglected potential more fully, a number of strategies are available. The visual disciplines, such as graphic design, illustration and animation have several processes which can be utilized. By borrowing these techniques, the developer can streamline the design process and make graphics production both more predictable and of higher quality. The use of production tools such as grids and storyboards, the appropriate use of colour and rendering techniques will therefore be discussed, and strategies for implementation shown.

This paper will survey how design procedures from related visual fields may be adapted to the benefit of the Computer Assisted Learning developer, and suggest that utilizing these methods may also feed back into the instructional design process, resulting in a better exploitation of the delivery medium.

Problem: Designing the CAL Screen

In everyday living, the human senses process the enormous amount of information transmitted by sensory stimuli. When working at a CAL terminal, the user's senses are also inundated with stimuli, but this time, some of them are specifically designed by the CAL author. In trying to make sense of the environment and draw some meaning from the sensory parade, the user tends to classify these stimuli into two groups: the important, which are generally referred to as signals, and the unimportant, or noise. Noise interferes with the reception of signals, and can often render a signal indecipherable. Instructional content must reside in that portion of the stimuli judged to be signals. Too frequently, the manner in which the content is presented results in it being perceived as noise; thereby not only concealing its own content, but also camouflaging adjacent material.

When a user is unable to copy with poor design, they may exhibit a variety of psychological responses such as the following (Foley and Wallace 1974).

Confusion: Detail overwhelms the perceived structure. Meaningful relationships are difficult to ascertain, and the conceptual model cannot be established.

Frustration: Results from an inability to easily convey one's intentions to the computer. It is heightened if an unexpected response cannot be undone or what really took place be determined.

Panic: Panic may be induced by unexpectedly long system delays. The usual response in this situation is to bang the key repeatedly, with the eventual result that some function is performed several times.

Boredom: This results from improper pacing and overly simplistic tasks.

These reactions frequently lead to physical responses (Eason 1979; Stewart; 1976) such as abandonment of the system; the system is rejected and some other source of information is relied upon. Another is incomplete use of the system, when only the tasks easiest to perform or most beneficial are employed. Historically, this is the most common reaction.

One organization surveyed its computer users, asking them what they looked for in the design of screens. The results follow:

1. A clean, orderly, clutter-free appearance
2. An obvious indication of what is being shown and what should be done with it.
3. Expected information where it should be.
4. A clear indication of what relates to what.
5. Plain, simple English
6. A simple way of finding what is in the system and how to get it out.
7. A clear indication of when an action could make a permanent change in the data or system operation.

Of these requests, the first four listed are all problems which require an exclusively graphic solution, and implementing the final three certainly involves a strong graphic component. The design of screens for any CRT then is a critical process, and that is especially true for CAL screens. Regular data entry screens can be learned in time by the experienced user, but the CAL screen may only be seen once by the student. One chance then, to separate the signal from the noise. One chance to discover the content and attempt to absorb or respond to it. And that one chance is so easily jeopardized by poor design.

The purpose of this paper is to introduce some techniques which have previously assisted visual designers in reaching good design solutions. For the CAL developer, this can only serve as a brief introduction to these topics. It is to be hoped that a better dialogue may then result between developers and visual designers.

Some of these processes are given with guidelines for use. It must be emphasized that these techniques can not only benefit the look of the finished product, but may also assist in the process of its development. Sound graphic procedures can contribute to budget and scheduling consistency, realistic expectations of CAL module size and improved communication between developer and designers.

We will be looking at four practices that can affect the design of the CAL screen: use of a grid, storyboard technique, the use of colour, and rendering.

Process: The Grid

There is a simple test for good screen design. When looking at the screen, can the viewer generally identify all the elements without reading? Do the parts of the screen have an identity independent of the text? This may seem to be difficult criteria to satisfy. And yet consider the front page of a newspaper. The reader knows where to look for the weather, can identify the headline and its following story and is able to locate the index for the remainder of the paper. This can be done with any newspaper, on any day.

This regularity and predictability is achieved through the strict use of a design grid. In graphic design, a grid is a template which sets the top, bottom, and side margins as well as the margins between columns of text or the space around pictures.

The grid establishes proportions on a page or on a screen. It gives order to information presented in a series, whether in magazines, newspapers, television or CAL. A grid becomes a consistent guide for the layout of a series of frames, because it is a style template giving the freedom to design quickly. The grid emphasizes predictability and decreases the number of variables involved in screen layout. It creates "windows of expectation" for the viewer, and hierarchies of importance in the material. The viewer soon knows where to look for necessary information. In CAL, where the designer frequently must keep track of screen overlays in all sizes and locations, the grid is an indispensible guide to the parameters involved in designing and placing these partial screens.

Most people working on CAL screen design do not have a graphic

design background, and thus are not fully aware of the many uses to which a grid may be put. More frequently, they are unconsciously influenced by the design formats used on television. TV uses an approach commonly known as "TV centred", that is the text content is almost always centred on the screen and kept well in from the edges of the screen.

This format convention results from physical restraints imposed by the early television picture tube and by idiosyncratic receivers. The early tubes were more circular in shape, but even as they have gradually evolved into the current rectangle, the area of sharpest focus (and therefore the logical place for text) has remained the centre of the screen. As the quality of television picture tubes improves, this convention is increasingly without technical merit, but remains a design habit.

Television producers also have to worry about cutoff; that is not all home TV sets show all of the broadcast image. A so-called "safe area" is a centred rectangle, measuring two-thirds of the width of the screen by two-thirds of the height.

The design convention of "TV centred" that developed around these two criteria is very much a part of our general visual vocabulary. Most people who sit down to design for a CRT (which does, after all, resemble a TV set) will automatically centre the text and keep it well in from the edge of screen. They have unknowingly employed a "TV centred" approach, and will insist that "it just looks right". This type of unconsciously learned design criteria must be recognized if a suitable CAL design solution is to be developed.

It must be added that this centred design approach can be a good choice under some circumstances. Care should be taken, however, that it be a specifically chosen solution, and not a result of cultural conditioning. If, for example, we were to attempt to recreate the functionality of the front page of the newspaper using a TV centred approach, we would most likely fail. Only a grid structure can support that kind of deep positional symbolism.

As general guidelines, we may say that a grid is well employed when a more "magazine-like" approach is needed; a strong integration of text and graphics is required; there are many possible screens; and there is a great deal of branching. The TV centred approach may be more appropriate in a game or simulation; when the content is mostly pictorial, with little text; when there are fewer screens; and if there is less branching.

Process: The Storyboard

A device commonly used in the CAL field is one borrowed from animation and film: the storyboard. In those industries, the

storyboard is used to plan precisely the complex sequences of action. It comes as a surprise to many moviegoers that virtually all the details of scenery, motion, camera path, and lighting are worked out months in advance of the actual filming. Alfred Hitchcock was famous for the complexity of the storyboards on which he planned his films. He went so far as to say that the filming and editing were quite boring, since the real creativity occurred during the storyboard stage.

Since within CAL, a well-conceived storyboard is a device used for both proposals and for production, the developer, in deciding on the storyboard format, must make some crucial tradeoffs between flexibility and accuracy. The storyboard must be flexible for design development and accurate enough for production purposes. The medium in which the storyboard is composed must be adaptable, easy to use, flexible, communicative and presentable. Usually, a storyboard is just precise enough to act as a vehicle for a proposal to the funding agency, and should really be called a presentation board. It is seldom reworked or tightened after the initial proposal and underdeveloped design ideas are thereby perpetuated.

But the storyboard can function as much more than a proposal device. As in the film industry, it is a communication device between programmers and designers; between the developer and designers; between instructional designer and content expert; and between the developer and funding agency.

The storyboard itself should be faithful to the objectives of the instructional design, and should be carefully composed in black and white first. If the design works, visually and pedagogically in monochrome, it will work in colour. Most CAL developers storyboard using traditional graphic methods, such as drafting tools, pencil and pen on paper, with a format drafted out and xeroxed for multiple use. This method is relatively easy, but subject to wishful thinking. It is always a temptation to believe that your content will easily fit on the target frame, even though a cursory character count would reveal that the planned screen is overcrowded.

Another storyboarding method has proved to be more successful. Ironically, the best device for visually planning computer assisted learning material is a computer. To be more specific, a capable graphics computer is the best storyboard tool. This graphics computer must have certain characteristics. It must have a resolution equal to or greater than the target machine, good quality printer output and easy-to-use graphics software, including both paint and vector drawing.

With a system such as this, it is possible to map a CAL delivery system's screen and constraints onto another computer's screen. All this is possible, of course, as long as the one being mapped

onto has the superior graphic capabilities. For example, it is relatively easy to map the ICON screen onto the Macintosh screen, but not to map the ICON onto the VIC 20. As mentioned, the storyboarding computer must have a resolution higher than or equal to the CAL delivery system.

In the accompanying diagrams (Fig 1-3), the basic text grid of the ICON delivery system was mapped onto the screen of the Apple Macintosh using Macdraw software. We can see how the text grid for the ICON's system fonts (Fig 1) underlies all the decisions made in laying out the following screens of this module. When composing the screens on the Macintosh, the background can either display, or hide, this text grid. Please note that the column headings shown (small uppercase letterforms) are customized graphic files, not system fonts.

Storyboarding on a graphics system such as this makes possible pixel-for-pixel accuracy. The size and volume of text can be reliably selected, and what you storyboard accurately reflects what you can achieve. Revisions are much easier and therefore more likely to be made. This ease of revision helps resolve the tradeoff between accuracy and flexibility. With a storyboard created on a graphic computer, you are less severely penalized for choosing a specific design direction than you would be were the storyboard composed using traditional graphic tools. As well, a storyboard created on a graphic computer can make for quality presentations and good, accurate communication.

In the CAL development process, the extraction of content from the content specialist, and the instructional redesign of that material into a deliverable form, is a lengthy process. Usually, the developer knows the target machine in advance but may then wish to format the material for multiple delivery systems. Or it may develop that the intended delivery system is to be technologically modified or completely changed.

It is easier to adapt to these changes with a storyboard that is faithful to the objectives of the instructional design and in a technologically flexible format. For these reasons alone, the practice of storyboarding on a graphics computer should be encouraged. With this method, the content is easily retained within the storyboard, and reformatting this material for another delivery mechanism is a far less painful process.

Process: Using Colour

The crude colour capability of most CAL workstations is frequently criticized. Most developers decry the number of hues available on the common delivery systems.

The colour capacity of upcoming delivery systems is being

improved greatly. In the computer graphics field, we have seen significant decreases in the cost of high end graphics systems. A typical setup, featuring an 8 bit frame buffer allowing 256 concurrent colours out of a total of 16.7 million has, in five years, dropped from over $125,000 to around $15,000. This kind of capability is becoming more common on machines which can serve as CAL workstations.

However, learning to compose with colour takes a visual designer many years of training; these skills are neither immediately acquired, nor enhanced, by contact with a powerful colour graphics machine. Learning to design for this level of colour ability, and to profitably employ the potentially powerful results, become important. This is a problem area, even for those well versed in colour theory. All artists and designers are familiar with subtractive or reflective colour, but few have experience with the properties of additive colour transmitted by the CRT. It is a new skill to learn and a new medium in which to compose.

For the CAL developer, the appropriate use of colour holds some wonderful possibilities. It also has some terrible pitfalls. In order to reach the former and avoid the latter, we can look at some general guidelines for the use of colour in screen design.

Simply stated, colour in CRT displays has two main functions: to differentiate and to draw the users's attention. Colour as a distinguishing mechanism can be used for these purposes:

1. To relate things, while ensuring that the similarities are intended.
2. To separate things, as long as the differences are planned.
3. To form groups and categories. Be certain that articles visually grouped by colour have a functional relationship.
4. To assist reading specific text information.

The designer may utilize colour to draw the user's attention for the following reasons:

1. To allow focusing. This involves directing the user, not visually forcing him. Try to employ harmonious colours for the basic palette.
2. To illustrate and identify. Consider the colour's symbolism and make sure that the visual code implicit in the use of that colour is known and relevant.
3. To entertain. Be very careful; overuse soon becomes boring and forgettable.
4. To move the user's eyes to the location of choice.

Just as there guidelines to the appropriate use of colour, there are cautions associated with its inappropriate use. Do not use

colour because it's there, for then it is gratuitous. Do not use a colour because you like it. Your favourite hue may not be the same one the user would choose to look at all day. Above all, do not use colour if it doesn't work; poor use of colour is much worse than no colour at all. Choosing not to use colour can often be the best choice. A texture or hatch pattern may be a much better technique to employ.

Always design in black and white first. If the design is monochrome, it will then work in colour. Everything in a screen design should be identifiable without use of a specific colour. Besides, the user may be colour blind. In one survey in the United States, 8 per cent of males and 0.4 per cent of females were found to be colour deficient in some form. When choosing a colour, consider the following:

1. Consistency, to permit the accurate formation of groups and categories.
2. Compatibility with the viewer's expectations, to permit the user to identify.
3. The degree of relevance to the subject matter.
4. The frequency of use of colour. Excessive colour on one frame can interfere with its intended use on subsequent screens. Choose subdued colours as the viewer will tire of these much less quickly.
5. Discriminability, to avoid confusion and assist perception.

The legibility of text on a computer screen has been the subject of much study. Insufficient or excessive contrast between the copy and the background can be detrimental. Through practice, it has been found that the most distinct body text is approximately 75 per cent more or less brilliant than the background. In other words, on a black screen, the easiest-to-read text would be a light grey, about 75 per cent of the way from black to white. On a white screen, the best legibility would be achieved with dark grey text, about 75 per cent of the distance from white to black. This is not an absolute rule. Because the human eye responds selectively to different wavelengths of light, different on-screen colours will vary in their adherence to this guideline.

In most circumstances, limit your palette to eight colours. Too many on-screen at once can prove very confusing. If you have access to more than eight colours at once, just choose your palette well and restrict the total number seen at any one time. Remember that colour's power to communicate is also its ability to confuse.

Process: Rendering

When an object is represented graphically on any medium, whether it be on a television, in a magazine, or on a computer screen,

the viewer of that image seldom compares it with the actual original object. An image of an object is, after all, two-dimensional rather than three-dimensional, occupies a different visual angle than did the original, and is surrounded by environmental stimuli of a different character.

What happens instead is that the image is evaluated and placed in context by comparing it to something in the memories of the viewer. The image is seen as a separate entity in its own right and the viewer seldom confuses it with reality. The graphic representation is not, after all, a copy or reproduction of the object. The function of a graphic is simply to suggest the appearance of the original. This prompts an association between the graphic representation now on the screen, or page, and the memory of the reference object. This distinction, that a graphic is, in most cases, a reference to memory, is an important one to keep in mind whenever considering the use of a "lifelike" illustration.

Even in cases where the illustration is of something not previously seen and the viewer has no reference object in his or her memory, a good graphic representation refers to conventions of depiction which relate to retained memories. Memory then, plays a significant role in the evaluation of that image, and care must be taken not to jolt that expectation.

Within the CAL environment, rendering is profitably employed in several ways. First, when the viewer is unlikely to have seen the object previously and needs a mental snapshot for later comparison. Secondly, when the information to be transferred is technical and specific, and object identification is necessary. Rendering can be useful if there is little animation involved; rendered objects have a solidity that inhibits the suspension of belief required by animation. A fourth instance occurs when the viewer may be able to see the object currently and make a direct comparison between the on-screen image and the object. Rendering should only be attempted when the visual designer and the delivery technology are up to the task. For example, it would be counterproductive to render the muscles of the heart at work; a rendering pales in comparison to the real thing. A task like this would be better done graphically.

Rendering is useful to make a game or simulation "lifelike". Rendering lends legitimacy to an image; it is often as being "real". More properly, it is often assumed that a "real" reference object must exist. Now, some games or simulations are better left with more abstract imagery; it all depends on the instructional design objectives. Overall, render when the object exists and a very specific visual reference to it is necessary.

There are many instances when rendering is an unsuitable approach. Rendering is unprofitably employed when the object

depicted had general cultural and conceptual references, i.e. a house. It should also be avoided when the graphic is an analogy or a part of a conceptual model. Rendering is inappropriate when the quality of the illustration interferes with the lesson flow or when there is animation. The rendering should not be used to impress the viewer. The rendered image carries meaning with it which should not be abused.

The rendered image is readily interpretable and it legitimizes its subject matter by lending an air of reality. The designer must be sure that he wants to impart this type of concrete interpretation to the viewer.

Summary

In this brief survey, we have seen that poor screen design can result in the incomplete use of a CAL delivery system, and that clarifying the screen is mostly a graphic task. There are several processes which can assist in this endeavour. Where applicable, the designer should employ grids for clarity and positional symbolism. Storyboarding, preferably with a graphics computer, should be in an accurate and flexible format. A quality storyboard retains the content should the delivery medium change. Colour is best used selectively, knowing that its power to communicate is also an ability to confuse. Realistic renderings can legitimize your subject matter, but care must be taken to ensure that the resulting interpretation is desirable.

A COMPUTER AIDED INSTRUCTION IN DYNAMICS, (CAID)

by: Hadi Farazdaghi
School of Engineering
University of Guelph

ABSTRACT
A computer aided instruction program is developed for particle kinematics to be used as a tutorial help for engineering students. The language employed for the program development is Prolog, an artificial intelligence language, which provides flexibility for information processing and facilitates further expansion of the program into other areas of dynamics.

The program provides information on classification of the subject area, the logical relationship of the subdivisions, and examples with three levels of help for each, during the tutorial session.

INTRODUCTION

The Computer Aided Instruction in Dynamics (CAID) is aimed at providing a tutorial guide for the study of dynamics as the first or introductory course. As a nucleus for this project an attempt is made to develop a computer program which can utilize modules of data and rules, and therefore be flexible and easily expandable as the project grows. The subject area covered by the present program is particle kinematics, and the language employed is Prolog, (MPROLOG release 2.1, 1986), with:

a) the capability of list processing for convenience in classification of information, and
b) modularity in data presentation for flexibility in expansion of the program.

Although the program is text-independent, and its coverage is compatible with those of the major text books, it utilizes examples from problems presented in " VECTOR MECHANICS FOR ENGINEERS" by Beer and Johnston, one of the leading textbooks in the field.

ANALYSIS OF KNOWLEDGE DOMAIN

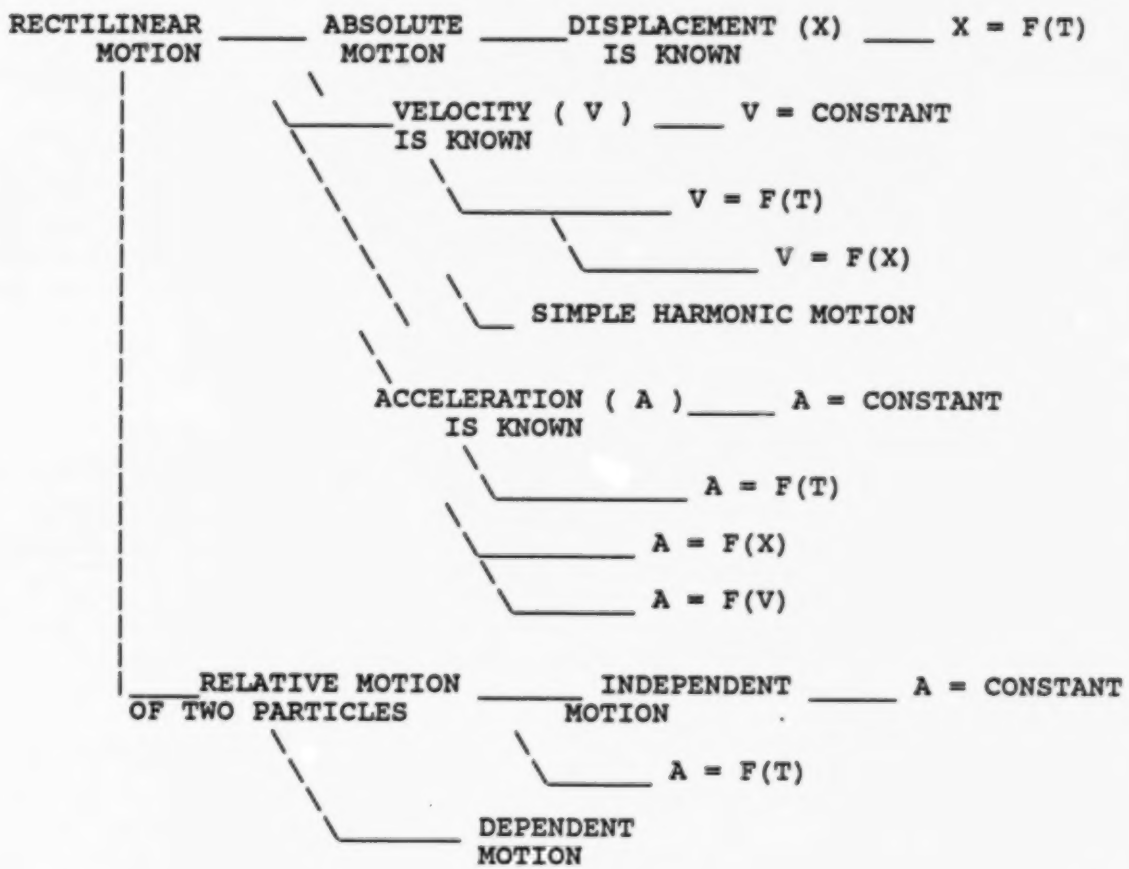
The area of kinematics covers the motion of bodies without regard to their size. This area is divided into two sections:

1- Rectilinear motion of particles,
2- Curvilinear motion of particles.

The method of problem solving is based on the nature of motion which may be either absolute or relative, the information which may be available for a given variable, and how that variable is related to others variables. Section one is subdivided, based on the above criteria, into either absolute or relative motions. For

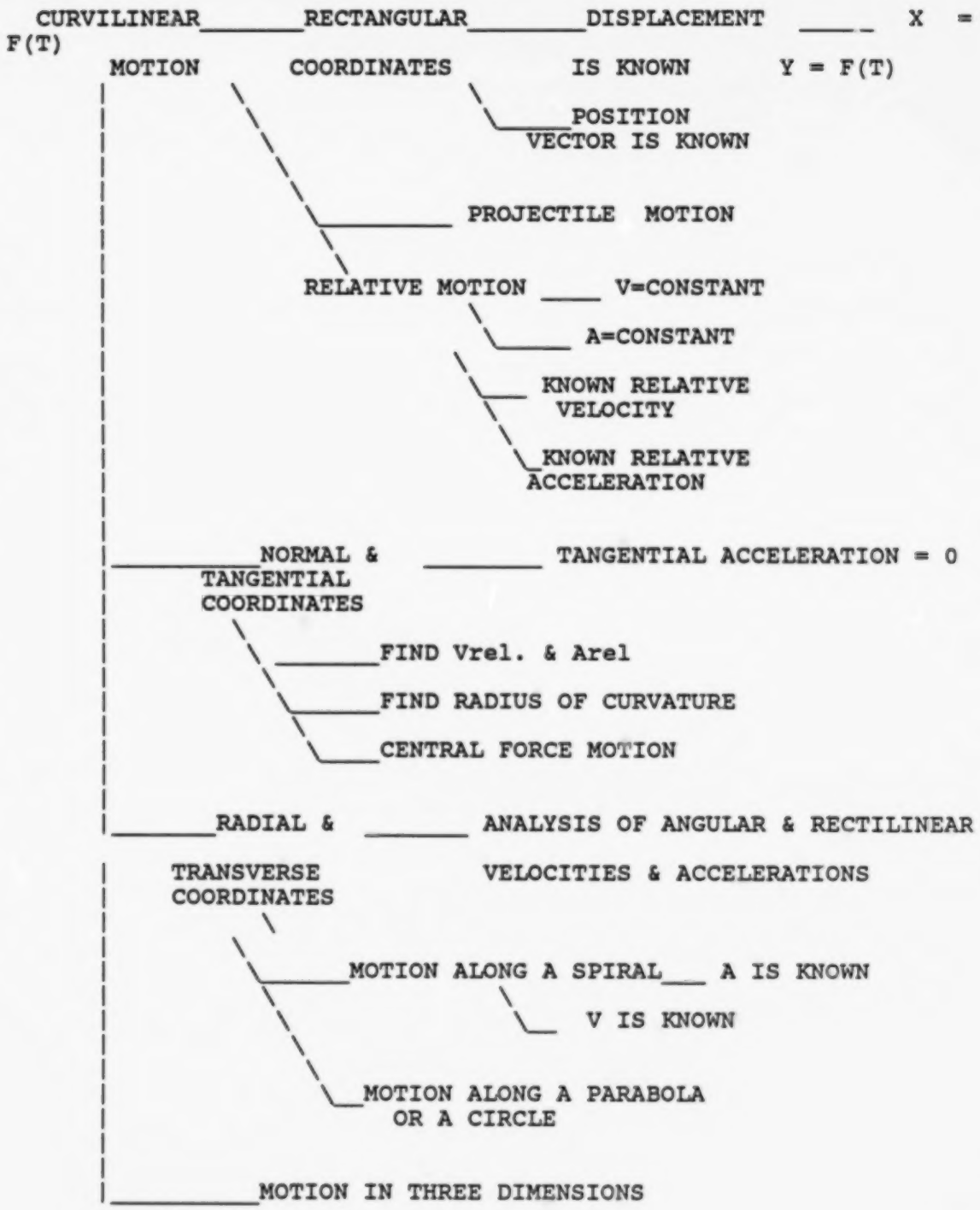
the absolute motion, three possible cases are considered, depending on which one of the three variables, displacement, velocity, or acceleration are known; and whether they are constants, functions of time, F(T), or one variable is dependent on another one. A rather different but appropriate classification is also made for relative motion. the tree structure resulting from this analysis for both absolute and relative motions are presented in Fig. 1.

Fig. 1. The Classification and Tree Structure of Knowledge in Rectilinear Motion of Particles



Similar analysis for classification of domain is made for curvilinear motion of particles, considering that, in this case, the coordinate system through which the motion is defined determines the direction of the analysis and problem solving. Three coordinate systems are considered, and the motion is analyzed according to Fig. 2.

Fig. 2. The tree structure of domain in curvilinear motion of particles.



THE STRUCTURE AND FUNCTION OF COMPUTER PROGRAM

The computer program consists of three separate modules covering screen selection menu, knowledge base rules, and data base facts respectively. Although the knowledge is presented in a general form which makes it possible to be used independent of any specific text, examples which are used in the program are taken from the 4th edition of "Vector Mechanics for Engineers", by Beer and Johnston, published by McGraw-Hill, 1984. This treatment provides an extra opportunity for the student to, if necessary, have a commonly available text for further consultations.

The program menu provides information for selection of the areas covered (chapters, at present kinematics of particle), sections (rectilinear and curvilinear motion), and the subsections shown in knowledge tree structure. Furthermore, two text specific types of information are also included in the menu which are the type of units (SI metric, or US), and problem numbers related to branch in the tree. Solution may be available for some problems. In most cases there are up to three levels of help for a problem, leading to the final answer. Inquiries can be made at any given level of the tree; for a problem number, areas covered in a section or subsection, problems related to those areas, or hints for solving an example problem available for a certain area. For example, an inquiry about problems on RADIAL & TRANSVERSE COORDINATES, ANALYSIS OF ANGULAR & RECTILINEAR VELOCITIES & ACCELERATIONS will result in a list of problem numbers with specific information on each problem, and an indication that problem 11.147 is provided with a solution. It is also possible to get a list of problems, for example, with SI units for a particular section. When a solution is provided it is assumed that the user is seeking general advice for solving the problem. Therefore, information is given according to the number of steps required in solving the problem, but not more than three steps, the last of which contains the answer to the problem.

A COMPUTER ASSISTED CURRICULUM PLANNING PROGRAM

BY: HADI FARAZDAGHI
SCHOOL OF ENGINEERING
UNIVERSITY OF GUELPH
GUELPH, ONTARIO, N1G 2W1
CANADA.

ABSTRACT.
A Computer Assisted Curriculum Planning (CACP) program is developed, to provide both general guide, and case-specific information for course selection in different semesters. The data concerning courses and their pre-requisites, and the rules governing the course selection are considered separately in different modules. An artificial intelligence language, Prolog (MPROLOG, Release 2.1), with list processing capability, is used in program development. The student's course file is compared with the course requirements of the academic program, and recommendations are made on the basis of course offerings in a semester, course pre-requisites, or corequisites.

INTRODUCTION.

The computer assisted curriculum planning (CACP) program is developed to help students in their course selection and program planning for different semesters of their studies. The objective of the program has been to minimize the time spent by students on the extraction of information from the university calendar, departmental guides, and other hard copy materials which are available to them.

The program is not meant to replace academic counselling, but it is expected that, by increasing efficiency, more attention can be diverted into other areas and components of academic counselling which can be more productive and fruitful. However, one major function of the program is to provide counselling assistance in academic program planning outside the office and work hours. The School of Engineering in Guelph has extensive computer facilities, including a network of microcomputers specified for free use of undergraduate students. These computers must not be used by graduate students and faculty, nor should any laboratory work be assigned to this facility. Students have access to the computer room without any restrictions. Furthermore, students are encouraged to have their own personal computers, and therefore, they are able to communicate with the undergraduate microcomputer network. Under these conditions it is felt that the Computer Assisted Curriculum Planning Program could be of significant help in providing the students with the required expert information on course selection without time restriction.

THE PROGRAMMING LANGUAGE:
Because of the diversity of courses, their prerequisites, co-requisites, and the rules governing course selection in an academic major, a program of this nature, may require a considerable amount of comparison between a large number of symbolic lists. However, such comparisons may not be best handled by the common procedural computer languages. Therefore, to avoid such limitations the language used here for program development has been PROLOG (MPROLOG Release 2.1), an artificial intelligence language which, because of its list processing features, can increase the efficiencies of both its programming and its application. Furthermore, as the rules of the program can be kept separate from the data base, future expansions or modifications to the program can be achieved with minimum alterations to the appropriate modules. This aspect is of particular significance, considering the dynamic nature of academic programs which should respond to the needs of the society in a rapidly changing technological age.

"CACP" is at present directed toward the requirements of the students in B.Sc.Agr. Mechanization major. However, the program may serve as either a base, or a model in development of curriculum guide for other academic majors. The program is composed of a number of facts, which make the data base, and rules, which govern the course selection as follows:

FACTS:
The course requirements of the major are divided into two groups, as follows:

1- Core courses, which are to be taken by all students;
2- Optional courses with some degree of flexibility for different students.

The core courses for 1987-88 academic year of the Agricultural Mechanization major are shown in Table 1.

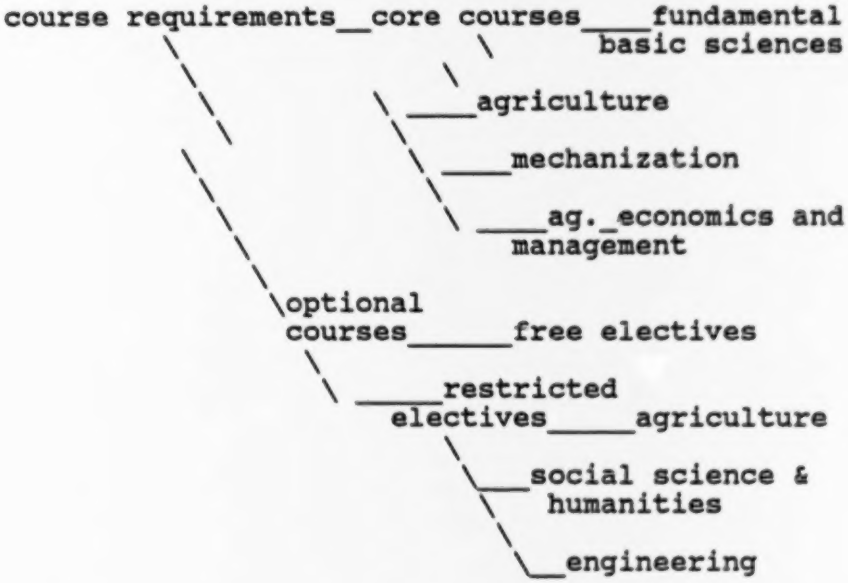
Table 1.The Core Courses for Major in Agricultural Mechanization

01-140	Agriculture & Food System I
01-150	Agriculture & Food System II
01-160	Agriculture & Food System III
01-200	Agriculture & Food System IV
02-222	Financial Accounting
02-223	Management Accounting
02-331	Operations Management
OR	
02-432	Financial Management
05-270	Mechanics
05-271	Graphics & Communication
05-272	Energy & energy Conversion
05-372	Materials of Construction
05-373	Fluids & Fluid Systems
05-374	Farm Buildings
05-375	Irrigation & Drainage
05-470	Electricity, Instrumentation & Controls
05-471	Mechanization Systems
05-473	Farm Mechanization
17-104	Botany
19-100	Fundamental Chemistry I
19-179	Organic Chemistry
27-120	Introduction to Computing in Agriculture
36-121	Introduction to Economics
37-109	Writing for Scientists
63-108	Elements of Calculus I
65-100	Microbiology I
76-108	Introductory Physics for Life Sciences
89-200	Statistical Methods I
92-102	Animal Kingdom

The program core courses are subdivided into "fundamental and basic sciences", "agriculture", "mechanization", "economics and business management". The "optional" or elective courses are grouped into "free electives" and "restricted" electives and the latter is further subdivided into "social science and

humanities", "agriculture", and "engineering" electives. The tree structure of this classification is shown in Figure 1.

Fig.1. The classification tree structure of Agricultural Mechanization courses.



RULES:
The rules and regulations governing the program may be classified as follows:

a- Entrance course requirements:
The following grade 13 or Ontario Academic Course (OAC) subjects are required for entry to the program:
1- One credit in mathematics;
2- Two credits from biology, chemistry, physics, and mathematics,
3- Two additional credits.

b- Order of course selection:
1-A course without prerequisites or co-requisites may be taken at any time.
2-A course with prerequisite may be taken if the prerequisites of that course are taken in previous semesters.
3-Corequisites of a course must be taken either in the same semester as the course if they are taken for the first time otherwise, they can also be treated as prerequisites.

c- Minor programs:
Minors in Agricultural Business and in Animal and Poultry Science are offered to students of B.Sc.(Agr.) Agricultural Mechanization Program. The minors require the completion of five or six courses in addition to the core courses required by the Agricultural Mechanization program.

d- Restricted Elective Courses:

1- Agriculture: Students must choose a three course sequence from one the following groups according to a course list:

a-Animal and Poultry Science
b-Soil Science
c-Crop Science
d-horticultural Science

2- Social Science & Humanities: A total of three courses must be taken from humanities and social sciences with at least one course from each area.

3- Engineering: Students must take at least one course from a given list of Engineering courses.

The program is designed to provide either a general or a case specific advice for students. The general help covers the cases for the students who enter the program in the fall semester, with no failures. A total of five courses per semester are considered as the normal course load for general recommendations. The prerequisites for core courses and some restricted electives are considered in the program, but such conditions for free electives and courses in the areas of agriculture, social science and humanities, because of their diversity, are not considered at present. However, if a list of electives which are most commonly taken by students is prepared, the additional conditions with regards to course prerequisites can be easily incorporated into the program.

In the general recommendations, the student can inquire, through the menu, about the recommended courses from each area or altogether, for each semester. For example an inquiry about the courses for the third semester will provide the following response:

27_120, 65_100, 01_200, 05_274, one_eng_elective,
and for the fourth semester:
89_200, 02_222, 05,272, 05_271, 37_109.
While a similar inquiry about only the fundamental and basic science courses for the third semester will be answered as follows:
27_120, 65_100.

Course recommendations, for each semester, are made on the basis of required courses which are offered and their order of priority. The difference between the number of required courses and the normal semester course load is assigned to restricted and free electives, in a manner to make the graduation possible in eight semesters. The program can also identify any other group of courses for each semester as shown in the example.

In the case-specific situations, the background of the student first should be written to a file, which will be read by the program. The number of courses desired for the semester will be asked from the student. Recommendations in this case are based on the courses which are offered in the semester for which the pre-requisites are already in the records of student file. The order of priority of courses are core courses first and optional courses second. Corequisites, if present, will have priority over other courses in the same rank.

The program is being expanded to cover the requirements of individual students with the same major and different possible minor programs from other departments. The program can be used as a model for other academic programs and disciplines.

RAPPI - COMMUNICATIONS NETWORKING IN THE CLASSROOM

Linda Hazzan
Project "Open Door"
University of Waterloo

INTRODUCTION

In December, 1986, in an attempt to create a structured learning environment in which both students and teachers might work, Project OPEN-DOOR was begun. With financial assistance from the Department of Communications in Ottawa and the Ministry of Education in Ontario, the RAPPI/OPENDOOR project links together approximately 80 elementary, secondary and vocational schools across Canada, Italy, France and the U.K. and allows them to communicate in a conferencing environment. The conferencing system -- called RAPPI -- allows students from diverse regional, language and cultural backgrounds to share ideas and information, enhance communication and language skills and develop technical expertise with a new computer system. Schools participate for a small fee and have unlimited access for the school year.

As a research project, RAPPI was created to answer several questions pertaining to computer telecommunications and computer-aided learning in the classroom: First, is a computer terminal a successful and adaptable medium for the transfer of ideas; second, can a networked conferencing system with the complexity and technical advancement to send a thought, idea, or even a picture across the globe, be operated by an eight year old child?

Furthermore, what features of the system aid or deter the user; third, and perhaps most important to an educator, what is the effectiveness of a mail/conferencing system like RAPPI as a viable and adaptable educational tool?

To answer these questions, it is necessary to study and assess some of the pedagogic and technical factors which determine success or failure for many schools. Surveys of the schools involved, and our observation of activity in the educational conferences on the system over the past several months have revealed many of the achievements and setbacks experienced by a lot of the schools. More importantly, however, has been the on-going development of computer networking and telecommunication techniques as teachers and students utilize the system, and realize its potential as an effective educational tool in the classroom.

ON-LINE CONFERENCING IN THE CLASSROOM
OPEN-DOOR

As a conference, OPEN-DOOR pairs schools with similar language and scholastic interests and provides a specific area in which to create and develop interactive classroom projects. At present, forty-six schools participate in conference OPEN DOOR and, as the following list illustrates, the physical distance between the schools is often quite vast:

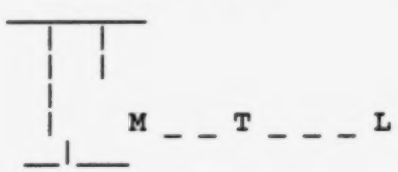
Bedford J.H. (N.S.)	&	Quebec H.S. (QUE)
Lycee E Fermi (ITALY)	&	Ecole J.H. LeClerc (QUE)
Lycee E Fermi (ITALY)	&	Ecole Secondaire Charlebois (ONT)
Lycee A.B. Sabin (ITALY)	&	College Saint Maurice (QUE)
St. Jerome's High (ONT)	&	Liceo Scientifico (ITALY)
Sooke Elementary (B.C.)	&	Christie Junior (ONT)
Queensbury E.S. (B.C.)	&	Sauris Consolidated (PEI)
"Les Genets" (FRANCE)	&	South Hull (QUE)
Philemon Wright (QUE)	&	New Westminster S.S. (B.C.)
Salisbury (ALBERTA)	&	South Wirral (U.K.)
Cloverdale E.S. (B.C.)	&	Riverview Junior High (N.S.)
Bishop Pinkham (ALBERTA)	&	Mosslands High (U.K.)
Woodfield Road (ONT)	&	Saltcoats School (SASK)
Jonathan (QUE)	&	Illeville sur Montfort (FRANCE)
Serres-Castet (FRANCE)	&	Ecole Jean de Brebeuf (QUE)
Guillaume Couture (QUE)	&	Ecole Paul Jean Toulet(FRANCE)
Jacques Buteux (QUE)	&	Cleveland Elementary (B.C.)
LaSalle H.S.(QUE)	&	Lycee A.B. Sabin (ITALY)
Lord Strathcona (ONT)	&	James W. Hennigan School (USA)
Rockway Mennonite (ONT)	&	Teignmouth High School (U.K.)
New Westminster S.S (BC)	&	Burleigh Secondary (U.K.)
Dartmouth School (N.S.)	&	Dartmouth School (U.K.)

These schools have undertaken a wide variety of projects ranging from historical and geographic studies to cultural and language exchanges. In most cases, aspects of their country -- climate, location, historical landmarks and events, and cultural pasttimes -- are detailed by the students as each learns of the other's world. Students get to know each other by name, and projects often develop between small groups within each classroom. Enthusiasm for and commitment to these projects is evident throughout the conference and, as seen below, some students often find new ways to ask questions and exchange information.

6417/9. Jonathan School Montreal, Que. Fri. Mar. 27/87
"RE-BONJOUR" LES AMIS DE LA FRANCE.
Voici vos derniers indices.
La derniere fois nous etions rendus sur le fleuve St.-Laurent.
A present vous allez chercher une ile assez grosse et tres popleuse.

Le nom de cette ile a huite (8) lettres. La premiere lettre est "M", la quatrieme lettre est "T" et la derniere lettre est "L".

Bonhomme Pendu



Si vous avez trouvez l'ile en question, vous devriez voir les villes
qui suivent: Outremont, Westmount, Verdun, LaSalle, St-Leonard, Lachine, Dorval, Pointe-Claire, Anjou, Pointe-Aux-Trembles et St-Laurent.
Et bien entendu,
Sophie Banville, Noelle Graton Tretreault

P.S. -- Nous attendons toujour VOS indices pour nous permettre de trouver Illeville-Sur-Montfort.

Other interesting and creative discussions have also been developed in conference OPEN-DOOR. Philomen Wright Regional High School in Hull, Quebec, and New Westminster Secondary School in New Westminster, B.C., for example, have begun a "Confederation Debate" in which the students assume the roles of key figures in the debate over Confederation in 1867; through a series of responses within the discussion, Sir John A. Macdonald, George Cartier, and even some noted Americans try to sway British Columbia in its decision to join Confederation.

6255/6. Philomen Wright Hull, Que Thu Mar 12/87
To the Rt. Hon. John A Macdonald We are growing increasingly irritated with your fedearl fumbling. We are assumed of positive treatment by Cartier and your associates: We receive nothing! Member of your Cabinet are off in the foot hills of New York skiing. We feel a week's lapse is desirable at this juncture. Perhaps you will contemplate the consequences of B.C. joining the U.S. during this interlude. We trust that when we resume dissisions, you will be in a position to offer us firm terms.
Yours sincerely
Governor Musgrave

6255/7. New Westminster New Westminster, B.C. Mon Mar 16/87
JOIN! JOIN! JOIN!...Wonderful people of British Columbia WE LOVE YOU!!! WE love the air in Victoria! Join the wonderful United States of America TODAY! It's a kaleidescope of discovery and imagination! A delight to the senses. We are the land of Mickey and Minnie, opportunity, and indoor toilets. You're missing all of the fun! Don't succumb to those sillywillies in Canada! come

join the party! Get with it, dudes! Life in the fast wagontrack! OK!! so, we're special! And you can be too! Be the best that you can be! Alors, les petits especes de cul-cul la praline, on va se joindre au centre du monde, ou on va rester enterre dans notre cerceuil Brittannique? 'Mon Doux! SO!!! We're decided, right? By this time next week, you can be a state
love(and I mean that),
Arnold (Mr. Party) Benedict

St. Jeromes High School in Kitchener, Ontario has been paired with Liceo Scientifico in Bologna, Italy for over a year and has attempted communication in Italian, French and English. In addition, they have had the opportunity to study the history and geography of both Canada and Italy, and to learn more about Italian and European culture. Just recently the two Roman Catholic schools have begun a project to discuss the history of separate schools in both countries, and to study the implications of the new private and public funding issues in Ontario:

6241/13. St. Jerome's H.S. Kitchener, Ont. Wed. Apr. 15/87
I am happy that the students from Italy are sending news of sports. I know that my students will be fascinated but I am going to start messages in this discussion about a topic of local historical interest.

The schools in our province are going to be fully funded by the government by the end of next year. This means that even though all students of this school are of one religion, that is, Roman Catholic, they will be funded equally with students of other religions. This is the first time that this will have happened and it is a very hot issue locally. Anyway, I am gettig ahead of mayself and my students. They will be in tomorrow with their research and their questions about who pays your bills.
Thanks for all of the great communication. The students will be thrilled. Mary Beam

Other conferences have been created which utilize the system in a wide variety of ways, all with varying results. The majority of schools have used the conferencing environment to exchange cultural information in an informal social structure.

SCHOOL-TO-SCHOOL -- "My worm died, but my cat is alive"
In conferences like SCHOOL-TO-SCHOOL, electronic "pen pal-ing" allows students to read and exchange letters which detail the contrasting lifestyles of children across the country and overseas:

From Pinewood Elementary School Cranbrook, B.C.
Hi, my name is Craig. What do you do in your spare time? Do you have a dog, cat, snake or worm? Well I do! Well, I let my snake go on November 7 86:11:07. It scared my mom. My worm died but my cat is alive and so is my dog. The colour of my cat is black with white spots and the my dog is black. Oh I am 10 years old

P.S. -- Please write back
from Craig.

From Blewitt Elementary Nelson, B.C.
My name is Brad Berikoff. I go to Blewitt Elementary School. My favourite thing about Blewitt is the girls. Can you tell me about Pinewood Elementary?

From Christie School Toronto, Ontario
Hi Sarah, it's Paul from Christie School. I am 11 years old. My hobbies are swimming, baseball, tennis, hockey. I have a big brother John. He is 16 years old. He is very strong.
He benchpresses me 30 times a day. Please write back about you and your family. Yours Truly,
Paul

EXPLORER/MOTION
Some conferences are purely information exchanges. Conference Explorer, for example, created by and for students in Grades 4 and 5, combines learning and fun by challenging students' knowledge of a wide range of Canadian trivia. MOTION was created to raise awareness of the Rick Hansen "Man in Motion" tour, and to share reactions to, and support for, wheel-chair atheletes and those who suffer from spinal cord injuries:

5868/4. Christie School Toronto, Ontario Thurs. Nov. 6/87
Dear People,
I think Rick Hanson is doing a good deed to the world. I mean riding all across the world. I really like what he is doing for us. By riding across the world he has proven that physically challenged people can do anything other people can do.
Maybe people will think different about disabled people.
Paul Sousa
Christie School, Toronto

WRITING/REWRITING -- "Harry the Flying Horse"
The system has also been used to promote communications skills. Writing/Rewriting, and several discussions in conference School-to-School, have elementary and secondary school students from different areas of the country working together on-line to write, edit and expand short stories. Teachers oversee the

writing and editing, but allow the students to incorporate their own changes. The introduction to conference WRITING/REWRITING outlines the usefulness of such exercises to students:

6331. Introduction to Conference Writing/Rewriting
The idea of this conference is for a student to post a ROUGH draft of a creative piece of writing. A designated student in the other school will rewrite the piece and then post the result.

It is based on the theory that students learn to write better through cooperation. By analyzing and revising each other's work, students are able to develop their writing skills and, in time, become better writers.

Teachers also like to encourage the collaborative style of writing at the elementary school level; however, rather than individual students trading pieces of work back and forth, whole classes of students write only part of a story -- a chapter or section -- and then post it for another class to read and continue. Harry the Flying Horse is just one of these "works-in-progress" which have been written by two grade 2 classes -- one in Sooke and one in Cranbrook, B.C. The following are two passages from that story:

5913/27 Sooke School Sooke, B.C. Thu. Jan 29/87
Harry the flying horse went to a race course. He had yellow eyes and purple wings. He also had blue horseshoes. One side of him was green and one side was pink. He went to the race-course in the afternoon.

The problem was he kept tripping over his hair. He fell down and he got up again. He was about 3 cms away from the finish line, he jumped up into the air and when he landed he got tangled in the reins.

Written by: Denise, Jennifer, Billy, Juergen, Shawn (Div. 10) We are looking forward to your class adding another chapter to the story.

5913/28. Pinewood E.S. Cranbrook, B.C. Mon. Feb. 2/87
While Harry was getting untangled, the other horses in the race passed him. He was very sad. He flew away to a distant meadow to cry. While he was crying, Freddy Fox came along.

"Why are you crying?" asked Freddy.
"Because I lost the race", said Harry. He wanted to become friends with Freddy.

They played together a long time in the meadow. They played football, dodgeball and tag. Harry was the best at tag because he flew in the air just when Freddy was going to catch him.

Because Freddy wanted to fly, Harry said he would help him build a pair of wings.

Written by: Andrea, Scott, Brandy, Tanya, Brent, Amy, Lucas & Lottie. Grade 2 Pinewood Elementary School.

HUMAN AND TECHNICAL FACTORS

Creating the environment for learning is only the first step in effectively untilizing a system like RAPPI. Many schools begin with great enthusiasm, but are hindered by technical and administrative obstacles. Problems with equipment, difficulty obtaining phone lines for modem connections, financial restrictions and time constraints are just some of the numerous setbacks which prove frustrating and demoralizing for teachers. As the following feedback from one teacher new to the system illustrates, patience and commitment are important elements in the learning and implementation of a system like RAPPI:

It has taken me a lot of time to figure out how this system works. In fact, we have spent the last few evenings trying to get this message to you. I think that in time we will be able to make better use of the system.

I now see why so many schools are actually inactive. This will take a lot of learning.

Many schools quickly overcome these obstacles only to find that the school with which they have been paired has not; so, anxious to begin their joint projects, they are often forced to delay, or even abandon, good projects because of an inactive or uncommitted partner.

Most schools are understanding and patient when faced with these difficulties, and will be supportive when problems and confusion arise. The following two messages -- addressed to RAPPI manager -- illustrate this point:

To: RAPPI MANAGER
From: Ken Cressman
Hi there. This is Ken Cressman. I put a response in [discussion] 6294. Hopefully, someone will answer soon. We'd also like to start talking to someone overseas (as would everybody, I suspect).

Did you hear about the sick bumblebee who kept doing his work while ill? Of course he spread disease as he staggered about his duties. The so-called "Blight of the Fumble-bee".

To: RAPPI MANAGER
From: Kevin Kroeker
Hello guys. I'm still trying to determine if and who we are paired with in OPEN-DOOR. Titles of the conference list us with NOTRE-DAME (no country specified), we have a list that says

we are paired with RIDDLESTON HIGH in the U.K., and I got a message from Ray (who saw my response in the OPEN-DOOR discussion between us and NOTRE DAME) indicating that he had three schools in mind that might be a good match for us.
The essence of this message is "HELP!"
Thanks much, Rockway Mennonite.

Interestingly, the majority of the existing difficulties are problems which teachers, not students, are having. Feedback from many schools suggests that the students who use the system -- from grade two to grade thirteen -- are enthusiastic and highly motivated about the projects undertaken.

Many of these organizational, technical and administrative difficulties can and will be overcome. However, the motivation of individual schools, and the degree of creativity, innovation and skill with which the system is used intelligently as an educational tool, can only come with the exchange of ideas between teachers, and the development of new telecommunications techniques in the classroom. (Here, too, the conferencing system can become an easy, inexpensive way to exchange information not only between students, but also between teachers.)

PRESENT AND FUTURE BENEFITS OF RAPPI

A global networking system has obvious educational benefits as geographic, cultural and historical information is exchanged. To learn about France from an atlas or a text book is worthwhile, but to open the classroom to a country by actually communicating with those who live there is an extraordinary experience for a student.

Unfortunately, the excitement over such technological capabilities has meant that many teachers have used it only as a glorified mailing system. Personal messages from one student to another are fun and interesting for the two involved in the exchange; however, the fact that the conferencing environment allows more than one person to read and respond to input means that such limited exchange does not put the system to its best possible use. Quickly the network becomes limited in scope, and the pen pal style of exchanging lifestyle and cultural interests becomes a narrow use of the technology.

LANGUAGE SKILLS -- The French Immersion Experience

With bilingual and French immersion schools spread across the country, the need for effective studies in that language has increased in English-speaking schools. Many French immersion teachers have found that students from English-speaking backgrounds have little opportunity tc utilize and practise their newly-acquired skills in their communities; they never use their skills from class in any other context. Thus, a sort of

"classroom French" develops where students converse only with their teachers and fellow classmates in an artificial atmosphere.

The linking of these students to native French speakers across the country and in Europe, means students are motivated to learn new vocabulary, develop their writing skills, and improve their command of their second language through contacts with interesting new peers. These students are not only motivated by the challenge and excitement of a new medium, but are also learning from each other in other areas of the curriculum (history, geography, etc.).

There are also many students who come from various ethnic backgrounds and speak their native language at home, yet never have the opportunity to use or learn it formally. Thus, their written skills in their "heritage language" are often lacking; many children are verbally fluent in languages like Italian, Portuguese and German, yet have virtually no written ability in that language. Many schools do not offer formal training in these languages and, therefore, there is no classroom facility to develop the child's second language skills.

Once again, the ability to link a student to a native speaker of a heritage language in another school, region or country allows a teacher to expand the classroom and to draw on resources not available locally. A system like RAPPI can give a student a dozen teachers and abundant opportunities to learn and develop new second language skills. Although projects such as these have yet to be developed and implemented on RAPPI, several schools have shown an interest in developing these "heritage language" skills with their students.

WRITING SKILLS

Teachers often find it difficult to motivate students to develop spelling and grammar skills. Ditto sheets and spelling drills are often tedious for both students and teachers, yet it is certainly agreed that these skills are important and necessary. Still, they are very often not stressed in order to promote creative written expression: Too much red on a page discourages spontaneity.

Systems designed for networked communications, and especially for computer conferencing, are ideal for encouraging these skills without one being denied in order to promote the other. As was illustrated earlier, projects like Writing/Rewriting encourage students to share ideas, and to write in a collaborative, cooperative atmosphere. In addition, students rather than teachers can be made to edit and comment upon their peers' work. Thus, students are often motivated to "perfect" their final drafts, knowing that they will be "published" and available for many eyes to see. Stories like Harry the Flying Horse go through

many drafts, from initial creative expression -- with many imaginative ideas and a corresponding number of grammatical and spelling errors -- to the final draft, published in a conference called (for example) Library, where students can see both their own work, and the work of their peers. Students will learn the necessary technical skills because they care about and understand the need for such precision.

COMMUNICATIONS SKILLS

Very closely connected with writing skills are communications skills. The ability to convey a thought or feeling in writing, without confusion or misunderstanding, is an ability which must be learned and exercised. In an age of computer communications, such an ability takes extra stress where one's communication is denied the expression of body language or intonation. The written word can easily become ambiguous, vague or even offensive in such circumstances so "exchange" teaches care and thought in the words of a message.

It is important to realize that what is said and what is meant can often be widely divergent. Seemingly innocent mail messages, with humour as their intent, are often miscontrued. One child who received a mail message written all in capital letters and with many exclamation marks replied to the sender, "I know, I know. Stop yelling at me."

Thus, a new kind of "computer etiquette" must be learned which is dependent upon the development of both writing and communications skills. In doing so -- in learning to communicate using the written word -- students at all levels will come to understand the practical importance of style, and to be aware of tone and audience as they write.

CONCLUSION

The potential for conferencing systems like RAPPI is vast and far-reaching, and the excitement over this technology is now beginning to grow. The involvement of schools in projects like RAPPI/OPEN-DOOR can be a vital part of computer education in the classroom as a wide variety of technical and creative skills are developed, and new ideas and experiences cultivated. The RAPPI/OPEN-DOOR project has been successful in coupling advanced computer technology with the imagination, creativity and expertise of both teachers and students, and has, in effect, opened elementary and secondary school classrooms to the world.

RAPPI - OPEN DOOR

Chris Hudel
Department of English
University of Waterloo

HISTORY

Rappi is a research project begun in 1985 by the Inter-Provincial Association for Telematics (IPAT) to exchange educational ideas and technology among schools in Canada, Italy, France, Great Britain and the USA.

Rappi's premise is that computer networking, by linking students in an interactive, inviting atmosphere with their counterparts in different backgrounds and cultures, will benefit all users in both education and technology. It has been my experience that suggestion and guidance, rather than direct prompts and orders, help students more quickly relate to one another -- primarily because they communicate on a level of their own understanding, not that of their teachers'.

Rappi's "home base" for the more than ninety-five sites around the world is the MTS-General Computer System at the University of British Columbia (UBC) in Canada. It is "overseen" by three people; Roger Hart, in Victoria, B.C. acts as our systems mentor and resolves the knottier technical problems. Linda Hazzan, an English Co-op student at the University of Waterloo, links schools and manages the bulk of user correspondence and enquiries for the "OPEN-DOOR" project within Rappi. I am a Computer Science/English Co-op student at St. Jerome's Collegiate in Kitchener and I handle most technical difficulties and other administrative tasks such as creating accounts, distributing funds and allocating disk space.

PRESENT

In accordance with our philosophy, Rappi attempts an unintimidating, unpretentious structure by three interactive pieces of software. The $MESSAGESYSTEM, a private electronic mail network, is designed for private contacts between teachers and site administrators. More general is *USERDIRECTORY which keeps a database of Rappi participants categorized by user-id, name and site information. The third, the *FORUM conferencing program -- the hub of Rappi's activity -- provides a public conference format for open, easily accessed discussions by teachers, students, and classes in each school. The topics, created by both teachers and students together, range over broad areas of educational concern including such subjects as early Canadian history, computers in the classroom, and creative story writing.

$MESSAGESYSTEM
For our needs, the $MESSAGESYSTEM is the most vital part of Rappi. Through it, we perform the bulk of our work. We use it as a global mail system to announce changes or give general information. One of the interesting advantages of a "global-send" option has been our implementation of the "Pet of the Week Award". Each week we announce an individual (or school) which has made an exemplary contribution to Rappi and we dedicate a joke in that group's honour. It is in this electronic post-office, that we give and receive help and suggestions on a personal basis.

*USERDIRECTORY
The *USERDIRECTORY allows any user to search for another user based on his name -- either full, short, or nick -- or user-id.

```
-+---------
  |
  | User types: *find bornon
  | Response:    "Gerald (G) Bornon", Owner=IPB1
  |           Busphone:    + 33 (35) 91 24 89 in France
  |           Busaddress:   Ecole Mixte Les Genets
  |                   Rue Andre Marie
  |                   Sainte Austreberthe
  |                   76570 par Pavilly, France
  | User types: *find id=ipcf
  | Response:    "Danielle Boutet", Owner=IPCF
  |           Busphone:    613-731-7212
  |           Busaddress:   Ecole secondaire Charlebois
  |                   Departement d'informatique
  |                   2525 Alta Vista drive
  |                   ottawa
  |                   ontario
 +---------
-
```

From an administrative viewpoint, however, a problem exists. Since the information has to be manually entered by the user, preferably at the time of first signing-on, -- though this takes only five minutes and is initialized by one command -- users hesitate or decline to do this, with resulting omissions and lapses. To date, only a little more than half of Rappi participants have entered information fully, despite constant reminders and persuasion.

*FORUM
Within the *FORUM conferencing program lies the heart of Rappi. Conferences, titles, discussions, and responses are the descending levels of the conferencing structure. Their order and role is best understood by analogy to a newspaper. Rappi is the complete "newspaper". Within it there are many

sections, here called "conferences". These correspond, more or less, to "activities" or large topics. Within each section of a newspaper there are headlines, articles and editorials. Similarly, within each conference, there are titles, discussions and responses.

The applications of the conferencing level are far reaching and are limited only by the users' interests and imaginations. For example, in the MOTION conference, students discuss Rick Hansen's Man-in-Motion world tour for spinal-cord research. From this conference, the students have learned of his daily difficulties in the sub-arctic temperatures across Ontario and Manitoba and the isolation of the Prairies. As well, the students have learned what spinal-cord research is, and have gained an appreciation for disble persons. For instance, this sentiment is clear in a student's remarks:

```
+---------
  |
  |      5868/4. Marilyn George  12:40 Thu Nov 6/86  14 lines
  |
  |         Dear people:
  |            I think Rick Hansen is
  |            doing a good deed to the world.
  |            I mean riding all across the world.
  |            I really like what he is doing for us.
  |            By riding across the world he has
  |            proven that physically challenged
  |            people can do anything other people
  |            can do.
  |           Maybe people will think different
  |            about disabled people.
  |               Paul.Sousa
  |               Christie School
  |               Toronto
  |
  +---------
```

(Man-in-Motion has been suggested by us as a topic of of Canadian (and world) concern. It is an "open" conference and invites comment from all schools and participants.)

"OPEN-DOOR"

Since January, Rappi has stressed an innovative project titled "OPEN-DOOR", which involves twinning two schools, usually remote from each other, sharing a common point of interest, into a conference where they discuss educational topics of appeal to each. All users of Rappi may "read" these discussions, but only the paired schools' students may

contribute to them. Other readers may contact the conference supervisor-teachers and request to add a comment or suggestion for the subject to take, but the conference topic remains with these two schools for development and interaction.

It was my intuition that a little more than 10% of the paired schools would actually achieve an on-going discussion. Amazingly, paired schools are continuing their commentaries at about a 75% rate of success. We have, however, been disappointed by the time taken or required by several schools to get "on-line"; this has ranged as high as three to five weeks, despite our efforts to aid them administratively. During these delays the "twin" has frequently become impatient and some of the students' initial enthusiasm has waved.

Through the research project of "OPEN-DOOR" it is our long-range hope that a cultural exchange of information may occur among schools from Canada, Great Britain, France, and Italy. Such an exchange has occurred and is evident in the discussions between St. Jerome's High School, in Kitchener, Ontario and Lycee E Fermi in Italy. After the students have discussed more current topics such as fashion, music and recreation, they settled for a major educational issue. In this case, strangely as it may seem, discussing governmental policy regarding education -- in particular, full-funding for minority schools.

+---------

| 6241/13. MARY BEAM 13:16 Wed Apr 15/87 18 lines
|
| Hi there, I'm so sorry that we have been gone for so long. I am now going to do something that is not good for a conference, but I am going to take two paths in this conference. I am happy that the students from Italy are sending news of sports. I know that my students will be fascinated but I am also going to start messages in this conference about a topic of local historic interest. The schools in our province are going to be fully funded by the government by the end of next year. This means that even though all students of this school are of one religion, that is, Roman Catholic, they will be funded equally with students of other religions. This is the first time that this will have happened and it is a very hot issue locally. Anyway, I am getting ahead of myself and the students. They will be in tomorrow with their research and their questions about who pays your bills. Thanks for all of the great communication. The students will be thrilled. Mary Beam

+---------

Another specific and successful example of the "OPEN-DOOR" project lies within the pairing between Pinewood Elementary and Sooke Elementary schools in British Columbia. The two schools have developed an interactive on-line story-writing section using the structure of the conference. Each school has taken turns creating a "chapter" of a "story". Within the past four months the two schools mutually produced three books, mostly fiction and fantasy, dealing with unusual characters such as flying horses and ghosts. On average, each book is the collective work of twelve to eighteen students, between the ages of eight and twelve years old. Ten more books are currently under production!

```
+---------
|
|     Funny Farm Hotel
|
|     Once upon a hot summer day,in the middle of
|     July,there was this "happy" married couple going to a
|     hotel for their 10th anniversary. They had tried to get
|     into many expensive hotels but they were all full .Too
|     bad.So George insisted that they go to this one BROKEN
|     down so called hotel all alone on one long street
|     saying that it wasn't all that bad.
|
|     At first they looked at each other in disgust
|     because hanging from the sign were 12 heads cut off by
|     the neck.But they were desperate.So they went up to the
|     door and knocked at the purple door .Nobody answered so
|     they walked in.  And to their surprise they saw several
|     weird people. A very old person approached them with a
|     nametag around her wrinkled neck...
|
+---------
```

"ADMINISTRATIVE"
Rappi shares the MTS-General Computing system at UBC with, among other users, two other educational networking projects sponsored by that province and the University of British Columbia; School's Network and Kids-to-Kids, which operate much as Rappi, but on a smaller, local scale. Also, Rappi shares participants with other projects, such as the seventy-five school EDAN project operated by TVO under the sponsorship of the Ministry of Education of Ontario and the TIME'S NETWORK in Great Britain. Because of Rappi's phenomenal expansion, and service to these various other tasks, we are occasionally limited by the MTS-G system at UBC. The main problem occurs while UBC is upgrading and backing-up their system. It is scheduled from between 4:00 - 7:00 A.M. P.S.T. which coincides with one of the best times for us to call, between 7:00 - 10:00 A.M. E.S.T.

From September of 1986, Rappi's participant list has grown nearly three-fold and we anticipate a total participant list of over one hundred and twenty-five schools by June 15th, the closing of Rappi for the summer months. Rappi is currently funded by the Department of COmmunications in Ottawa and the Ministry of Education for Ontario which pay for the computer-time, DATAPAC connection fees and some personal costs. Because Rappi is 'free' to all its participants -- the only expenses being ones of time, participation, and a terminal/modem configuration -- we receive a great number of inquiries and participants. Unfortunately, possibly because it is free, many users are not as actively involved in the project as they would be if there were a cost. Because of this, records of active, inactive and "slumbering" schools are hard to maintain. On occasion, we will pair two schools in the OPEN-DOOR project on the basis of their stated interests, only to find subsequently that they have difficulty speaking the same language, or that we've paired a senior secondary school with an elementary one and the sophistication of the one works against the other.

FUTURE

If we were to conceive an enhanced educational network or could be granted a 'wish-list' for the future, these are areas that would require development. First, we would like to introduce digitized graphic and sound images as enhancements to the discussions. By doing this, the images of sight and sound may help to convey a complicated thought or idea. Second, it is out hope to further develop the $MESSAGESYSTEM into a separate conferencing system, where a group of individuals can talk among themselves as if they were participating in a teleconference, similar to COMPUSERVE's CB-Radio option.

There is, however, one major stumbling block to overcome; terminal configuration. Because Rappi is international, not all users can ever be presumed to use the same terminal. While most participants use APPLE, COMMODORE, IBM, or ICON systems, there is not one unified system for Rappi and this makes graphic and sound presentations impossible at present.

In the fall, when Rappi commences for another school year, it is our hope to institute a better form of organization for the Rappi Administrators. First, we would like to make the *USERDIRECTORY mandatory on the first sign-on to keep our records current. Second, it is our desire to formulate a standard of telecommunication among the participants to enable some full-screen editing and limited graphic transfers.

We fully anticipate Rappi to maintain an active participant list of slightly over two hundred. Overall, Rappi will be introduced to more rural areas and will include such conferences as Constructive Essay Writing, Computer Aided Learning and will run other educational pieces of software.

C. Stuart Hunter,
Department of English,
University of Guelph,
GUELPH, Ontario,
Canada, N1G 2W1.

COMPUTER CONFERENCING SYSTEMS:
Educational Communications Facilitators

Computer conferencing systems -- such as the University of Guelph's CoSy system -- provide the instructor with numerous ways of enriching the educational process through the use of current information technology. In a very real sense, computer conferencing provides the instructor and his class with a meetingplace that has no physical or temporal boundaries. It also enables the student and instructor to build an on-going, dynamic database of course-related material that is constantly available and constantly current. Beyond these obvious advantages, though, the most pedagogically useful aspect of computer conferencing is the way in which it facilitates communication and interaction both between instructor and student and between students in the course. Through the use of computer conferencing, the instructor has the opportunity to maintain a much closer, much more direct, and much more pedagogically efficient contact with students while the students not only have that same contact with the instructor but also with each other. Through computer conferencing one can truly achieve the boundary-less classroom.

The CoSy system is composed of three component subsystems: a MAIL facility, a CONVERSATION facility, and a CONFERENCE facility. The MAIL system functions as a standard electronic mail system, enabling the user to send and receive mail message to and from other users of the system. MAIL enables the user to send confidential messages to a single user or list of users on the system. MAIL can also be filed and retained for future reference. The CONVERSATION facility permits any number of users to engage in a single-topic, linear discussion of a particular issue or problem. The CONVERSATION originator establishes the ground rules for the particular conversation and then creates a list of members. Each member of the conversation can then read from and write to the conversation. Conversation messages can also be filed or can be transferred either to mail messages or to the conferencing system. The CONFERENCE system expands on the capabilities of the CONVERSATION system to provide a multi-topic continuing discussion of a variety of related topics linked by a central concern. Conferences can be open to all users of the system, or closed and therefore available only to invited participants. Conferences, further, can structure a discussion in a particular way since one has two ways of writing to a conference. One can also search a conference for occurrences of particular words and phrases. MAIL, CONVERSATION, CONFERENCE: the

three facilities of the system enable an instructor to facilitate intercommunication in a group or class to a remarkable degree.
System application:

The CoSy MAIL system can be used in three basic ways. It can act as an electronic alternative to office hours in that it provides a direct, one-to-one communication between instructor and student. By using the MAIL facilities of CoSy, a student can contact an instructor at any time and can receive comments and counselling virtually any time of the day or night. CoSy MAIL can also be used to submit and to receive assignments. At any stage in the composition process, the student can transfer an assignment to the instructor as a mail message and receive prompt and thorough commentary by the same route. Mail logs will clearly date-stamp the files, thereby removing any argument about the time at which the assignment was submitted or the fact that it was or was not sent or received. Finally, and most importantly, CoSy MAIL provides a way for the student to remain in direct and constant contact with his or her peers. Through CoSy MAIL, students can continue the discussion that was started in a face-to-face class, or can go on to discuss matters that only receive their initial impetus from the classroom material. CoSy MAIL, then, is a facility that goes a long way toward enabling us to build the boundary-less classroom. Yet MAIL is just one of three facilities that are available.

The CoSy CONVERSATION system, similarly, provides a means of communication; but whereas the mail system is designed for one-to-one communication, the CONVERSATION provides the user with an ideal multi-party brain-storming area. Students will set up conversations on the system in order to develop seminars. The originator will enroll the other members of the seminar team and then open the conversation with a comment about the seminar topic. The group will then discuss the topic at will, signing on whenever they are able and contributing what they are able. The final product of the deliberations can then be downloaded from the system to a micro for word processing and hard copy production or can be edited in the CoSy system and then transferred to the CONFERENCE system for the general class audience. As well as a seminar developer, the CONVERSATION mode also functions well for brief discussions among a small group of interested participants. Students who were engaging in a debate in the face-to-face class meetings have been known to then set up conversations to continue the debate after the class ends. Not of interest to the class as a whole, the topic of the conversation is still of interest to the individuals involved. Often too, one find that the conversation gradually produces a focussed thesis that can then be turned into a full-blown essay on the topic. The problem with the CONVERSATION mode, though, is that it cannot be subdivided into topics. It is singular in focus and linear in organization. While these factors are no drawback for uses such as seminar development, they are somewhat limiting if one wishes

to explore the various ramifications of a topic. As a result one finds that, often, participants in a conversation will eventually transform that conversation into a full-blown conference.

The CONFERENCE mode provides the electronic adjunct to the face-to-face class meeting. Originated by the instructor, the course conference becomes a repository for course-related materials and discussion. Topics can be established to parallel the content divisions of the course and/or to answer particular needs. A course in Renaissance Literature, for instance, may have a topic for each section on the syllabus as well as topics for assignments, examinations, and general discussion.[1] General announcements of interest to the class can be posted in the conference; lecture notes and other material related to course content can be posted; questions keyed to lectures or to course material can form the basis for discussion; sample essays and assignments from students can provide both a talking point and a means of improving student writing. Also, too, the conference provides an on-going, dynamic data base of course-related materials. Comments and questions from students can be used to develop a series of "helpful hints" messages. Comments from individual students -- received either in the conference or in mail messages -- can be used to form the basis for a number of tutorial strategies; questions and comments can be used as a means of altering and revising the course for future use. Generally speaking, the CONFERENCE facility enables the instructor to "cover the course content" fairly rapidly, so that more time is available for discussion and communication, the basic factors in education.

System potential:
The remarks above suggest some of the ways in which a computer conferencing system like CoSy can be used to facilitate the educational process. What has been discovered in using the system at the University of Guelph is that while the CONFERENCE and CONVERSATION modules would seem to be the most important from an instructor's point of view, it is the MAIL system that has been most effective as a pedagogic tool. Students use the MAIL system as a way of contacting their instructors; but more often than not, they are using it to contact their peers and to use the system as a way of carrying on a one-to-one discussion of a critical issue. Potentially, then, the MAIL system seems to have the greatest promise, in that the electronic message-swapping that goes on seems to act as a stimulus to the learning process.

Preliminary experience would seem to indicate that both the CONVERSATION and the CONFERENCE systems would be better used if one were to stop using them as substitutes for the printed text and begin to use them for their interactive potential. Using the conference as a repository for lecture notes is a poor use of the facility; using the conference as a repository of key discussion questions, on the other hand, seems to work extremely well.

Using the conference as a repository for trial examination questions, and then using the answers to those questions as a way of clarifying the pedagogical goals of the course also proves extremely useful and pedagogically efficacious.

The potential uses of computer conferencing in education are limited, in fact, only by the imagination of the educators. As Murray Turoff and Starr Roxanne Hiltz have shown,[2] the possibilities for on-line education are vast. What it is going to take to realize the potential of such systems is the active participation of individuals who are not afraid to imagine worlds we have never seen before: classrooms without textbooks, electronic blackboards, remote access education. We are going to need people who have a sound grasp of the principles of pedagogy and a willingness to embrace the possibilities of the new technologies to take us farther than we have ever dreamed of going. The possibilities are there.

Conclusions

Computer conferencing systems provide the student with a means of being in touch with instructor and peers 24 hours a day, seven days a week. Because the conference is always there, the student is not constrained, either by class hour or office hour, to be in a particular place at a particular time. Rather, student and instructor can meet on the system at their individual convenience. The instructor can send and receive mail messages, participate in conversations, and read and write material to the conference system whether (s)he is on campus or miles away. The student can sign on to the system whenever it is convenient and make contributions. Experience shows, though, that what makes CoSy the ideal boundary-less classroom is the fact that it provides for the student and the instructor a free form, fully-available communications system whereby contact can be maintained at any time. Access to the learning tools is one of the major requirements of a pedagogical system: CoSy provides that access.[3]

NOTES

[1] A "common room" topic in the course conference is often a very useful item, since it provides a place for students to discuss general course-related concerns in a free and open manner.

[2] See in particular Starr Roxanne HILTZ and Murray TUROFF, The Network Nation : Human Communication Via Computer (Reading, MA; Don Mills, Ont. : Addison-Wesley Pub. Co., 1978) and Starr Roxanne HILTZ, Online communities : a case study of the office of the future (Norwood, NJ : Ablex, 1984).

[3] For information on the University of Guelph's CoSy system, call the CoSy Development Team at 519-824-4120, ext. 3775. A history of the early development of the system is available in Judith C. Van Duren, Development and Early Applications of Cosy at the University of Guelph. (M.A. Thesis) GUELPH: School of Graduate Studies, University of Guelph, 1986. Reference material on the system is available in M. Brochet, et al., CoSy Computer Conferencing System Reference Manual. Guelph: CoSy Systems Group, Institute of Computer Science, University of Guelph, 1986, and in CoSy Computer Conferencing System User's Guide: Version 3.1. Guelph: CoSy Systems Group, Institute of Computer Science, University of Guelph, 1985.

Interactive Computer Art
Michael LeBlanc
Department of Fine Art
University of Guelph

Interactive Computer Art was devised to give Arts students (Fine Art, English, History and Philosophy majors) an introduction to information design. The creative arts are very much concerned with design-- Fine Arts students are familiar with visual design and visual thought. Writers design with words. Interactive Computer Art enables creative students from several disciplines to work in a new media made possible by the microcomputer. This new media is a combination of graphics, text and time, held together by the instrument (the computer). It is therefore possible, using the computer, to express ideas in different ways.

The class projects are designed with 'process' in mind. Where the student ends up is less important than the design strategy used to get there. This document is an account of the first offering of Interactive Computer Art, offered by the Department of Fine Art as an intensive 6-week session. Classes were set for Monday to Thursday, 6pm to 9pm, for the six weeks. Computer time was available during the daytime.

Funds for the course were supplied by the Department of Continuing Education, and computer facilities, support and classroom space was provided by the Instructional Technology Support Group, Office for Educational Practice.

Space was made available for 10 students, but 8 were finally admitted. Despite the advertising done in the History, Philosophy and English departments, only one non-Fine Art major applied. All were in their final (4th) year of study.

PROJECT ONE EVALUATION

OBJECTIVE:
To learn how to use the VITAL Frame Creator;
To learn how NAPLPS pages are constructed.
To learn how to edit the created pages.

RESOURCES:
3 hours Instructional Technology Support Group (ITSG) help
9 hours total class time
4 VITAL workstations during class time (including one workstation equipped with digitising tablet)
2 VITAL workstations (for sign-out during office hours)

PROJECT SCHEDULE:
August 8:
ITSG staff gave a demonstration on the VITAL Frame Creator and

gave each student a copy of the Tutorial exercise which consisted of a sailboat drawing composed of lines, arcs, polygons, fill patterns and text. The students were required to pair up into groups of two and complete the exercise by the end of the class.

August 9, 13 (group 1), August 14, 15 (group 2):
Magazine Advertisement (two day project); to convert a magazine advertisement into a NAPLPS page, taking into consideration the decrease in resolution, aspect ratio, and colour range available in VITAL over the paper copy.

Project Results:
Most students found the Frame Creator quite straightforward (all had a Fine Art background). Some were able to finish within the 6 hours of class time. Most required out-of-class time on the equipment. Two were able to complete 2 pages, though they deliberately chose very simple graphics for the first image.

The problems encountered:
Num Lock Key:
All students had problems with the Num Lock key when they chose to use the text editor. If the Num Lock was 'on' (and there is no way of knowing if it is on or off when using the IBM-PC) use of the keypad for cursor movement resulted in the overstriking of text with numbers.

Typing:
Students who are not typists were slower to learn the system.

Two Screens:
Many students seemed to be confused by the two display screens and the keyboard. It took several hours for some students to learn where to look for the necessary information.

Tablet:
The graphics tablet was very useful in some cases, but occasional failures in the operation of the tablet caused mis-entry and frustration for the user. In one such case, the computer was turned off and re-booted-- the tablet's operation improved significantly. (It could be that the software driver for the tablet is becoming corrupted during use over a 12-hour day).

Recommendations:
1/ All students should receive s short lecture on NAPLPS drawing strategies to prepare them for the Magazine Advertisement project.
2/ All workstations should be equipped with a non-keyboard entry device. At least one should be an 'absolute-positioning' device like a tablet. The others could be mouse-equipped.
3/ Other frame-creation software should be employed which makes use of a pointing device other than the keypad and uses one, rather than two display screens.

PROJECT TWO EVALUATION:

OBJECTIVE:
Working in groups of four, each group will create an interactive module in VITAL based on the story of the Three Little Pigs. The module will be able to successfully guide the participant through three different versions of the story.

RESOURCES:
3 hours ITSG staff help
21 hours class time per group
4 VITAL workstations during class time (including one workstation equipped with digitising tablet)
2 VITAL workstations (for sign-out during office hours)

PROJECT SCHEDULE:
July 9 (Group 2) July 14 (Group 1)
Three Little Pigs Story
Each group was instructed to 'brainstorm' to create three versions of the story. One member of the group was assigned the role of 'moderator' and 'secretary'.
Group 2's experience on this day was very fruitful, with remarkable cohesion between participants. The assigned work was completed within the session.
Group 1's interaction was more complex. The only male student in this group had some very specific ideas about how the module should run, and had done much preparatory work to support his ideas. The three females felt perhaps a little intimidated by this, and as a result the atmosphere for creative thought was ruined. It was decided unanimously between the instructor and group members that Rudy should create a module on his own, but that all group members should share graphics.
The three stories were not completed in the alloted three hours by this group.
From this point on, the instructor referred to the new groupings as "The Gang of Four", "The Gang of Three" and "The Gang of One."

July 13 (Gang of Four) July 15 (Gang of Three, Gang of One)
Storyboard
Using ITSG Storyboard forms, each VITAL frame was sketched in to give visual substance to the three stories.
For the most part, this session went straightforwardly, but the Gang of Three continued to lag behind.

July 20: VITAL Reference Tutorial Session
ITSG staff provided help in explaining how to create a reference module in VITAL, using the Tutorial Manual for hands-on training. This session ran successfully until it was discovered that one of the later versions of VITAL which was in use had an altered menu structure from that described in the Tutorial Manual. Confidence in the system was lost at this point among some students.

July 21-30 (All groups)

Frame Creation/Flowcharts
During this period, the Gang of Four alternated with the other two groups for computer access. At times when computers were not available, stories were revised, flowcharts were completed and other forms of project management were practised.
Internally, each group had its own way of assigning computer graphics work to its members:
The Gang of Four decided, in the interest of consistency, to have the same artist do all the wolves, another do all the pigs, and the other two artists tackle backgrounds. As can be seen in the VITAL module, this strategy was surprisingly successful.
The Gang of Three decided to apportion the work according to story, so that in this VITAL module each story has a different 'look.'

July 30
Routing the Module
It became apparent on this day that the project would not be completed in time for the Open House on the 5th of August. Routing took place for only those frames which could be linked to provide contiguous areas within the modules. Any 'dead ends' were covered by an "Under Construction" frame.

August 4
Testing and Debugging
Final revisions were made on the provisional modules.

August 5
Open House
This day was provided to show faculty and students the work-in-progress from 3:30pm to 8:00pm. At that time the doors were closed, pizza and wine was delivered and each group presented its module to the class for constructive criticism. Spelling and grammar mistakes were identified, and suggestions were made to clarify text, graphics and meaning.
The date for the Open House was fixed by external factors: Regular Spring Semester classes finish at this time, and what few administrators and faculty who were on campus were likely to disappear within the week.

OTHER ACTIVITIES from July 16 to August 13
Two Macintosh computers were made available with the programs "Guide" and "MacPaint." On July 16, all students were given a tutorial on the Macintosh and instructions on how to use MacPaint.
After a short time of play with MacPaint, the students were reassembled for instruction on the concept of hypertext and the use of the hypertext program "Guide."

The Guide Hypertext assignment consisted of a short text poem called "Mystery." The students were asked to find an interesting phrase or word in the poem, and then expand on it. In Hypertext this entails selecting the word or phrase and transforming it into a 'button.' The button can then hold an expanded text or graphics passage underneath itself. This augmented passage his hidden from the viewer until the button is clicked. In this way a single idea can be linked to other ideas in an intuitive manner.

Some students worked on the Mystery project, but because of pressures from the Three Little Pigs project, not everyone had time to make a contribution. Mystery was also featured at the Open House.

Some students managed to find time to work with a paint program on one of the PC's at ITSG called "Dr. Halo II." This mouse-driven program enabled the students to 'paint' on the screen using 256 colours and high resolution. Several pictures were produced using this program and were on display during the Open House.

August 13
Last Class day.
Groups were actually given until August 14 to hand in their Three Little Pigs modules.

ASSESSMENTS:
Individual evaluations were based on the level of understanding and utilization of the concepts dealt with in the course. Marks ranged from 74% to 80%. The highest mark was given to a double-major student in Fine Art/Philosophy.

The VITAL Modules

Gang of Four "Three Little Pigs Great Adventure":
This module was, in a large sense, designed like an interactive computer game. It began with an objective:
"How would you like to drink with a bunch of PIGS?"
It proceeded through three different temporal ranges: past, present and future.

The Past story put the participant in the wolf's shoes, with a crew of pigs about to mutiny.

The Present story switches first person to the pigs-- to IranGate hearings where the participant is asked questions by an intimidating wolf and no matter response is given the participant is accused of lying. The pigs escape custody by a remarkable fluke but are chased by the wolf to the White House, the Jungle House in Africa or the Sand House in Iran.

The Future story presents a situation where both wolf and pigs must co-operate to defuse the dreaded Hyperbomb. Depending on which wire is cut, the participant either saves the world (and gets a free beer with the authors) or the unthinkable happens. Because each group member created only partial elements of each frame, the pictures had to be assembled as multiple overlays before they could be loaded into the module. It was extraordinary that, except for one frame, all pictures came together as though all the parts were created at the same time. Careful planning and adherence to the storyboard aided in the assembly process.

Gang of Three "The Three Pigs"
This module was originally conceived as a takeoff on the murder mystery game 'Clue,' but borrows heavily from the popular evening soaps. As a result, it's a muddled mixture of pig sex, money and intrigue. In contrast to the ordered flow of "The Three Pigs Great Adventure," "The Three Pigs" sometimes jerks the viewer around the module for inexplicable reasons.

This group's lack of success may be attributed to the more individualistic temperaments involved. Because all three group members are dedicated artists (one of whom has chosen computer graphics as a career) the poor group performance did not appreciably alter individual marks.

Gang of One "Porkmeat"
This module is still not finished (it is being completed as of this writing) with much of the graphics missing. It is entertaining even without the pictures, however.
"Porkmeat" is the most interactive of all the works, with one decision every frame.

GENERAL COMMENTS
One of the problems with an intensive 6-week course is course conflicts with regular session classes. Final exams for the regular session often took time away from my students at the worst possible time. This would not have been so bad if the students were not working in groups, but there were evenings when work halted because one student in the group had an evening exam.

Absences due to sickness (in this case the flu) kept one student away for 6 hours of class time in two days. A regular session course is not so intensive, so fewer class hours would have been missed.

On the other hand, it was easier to borrow computer equipment for a six-week period than for the regular 13-week session.

EQUIPMENT AND SOFTWARE
The hardware performed well for the most part. A borrowed PC monitor died and had to be replaced late in the course.

VITAL, for all its quirks and inconsistencies, is a simple program which does what it is designed to do: provide authoring tools at the minimum of training cost. There were disappointments, however... I lost a student's Magazine Advertisement when I pressed a wrong key while loading files into a special module. The picture was destroyed without warning.

The Macintosh provided a 'break' for students who were relieved to be able to draw pictures without touching the keyboard. Most found the Mac's lack of colour to be compensated by its clarity and resolution.

INTERACTIVE COMPUTER ART-- conclusion
This course was a remarkable success. ICA helped Fine Art students to see past their familiar visual design training to the larger world of information design.

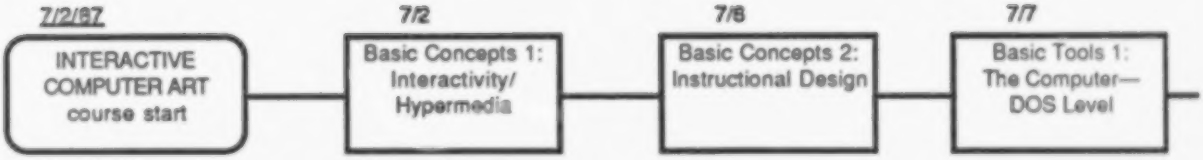
I believe many Arts students are born information designers. The university should offer a regular interdisciplinary course which deals with informational design. Not only would such a course be in line with the University of Guelph's expressed intent to provide leadership in information technology, but it would also provide practical preparation for a career in this field. (Sheridan College offers diploma courses in Instructional Design and its graduates report the highest placement and starting salary rate of any of its programs.)

One already-graduated student is currently shopping for computer graphics equipment. Three others are pursuing further studies with this instructor to work with VITAL or other more sophisticated authoring systems at ITSG.

Final say should be given to the students who completed the course. In answer to a questionnaire "Has this course changed your career expectations?" out of eight responses, one equivocated. The rest replied "YES."

APPENDIX

Interactive Computer Art 13-250



Course Goals:
Interactive Computer Art is an interdisciplinary course for undergraduates in Fine Art, English, History and Philosophy programs.

Students enrolled in the course use computers to creatively apply concepts and skills in:

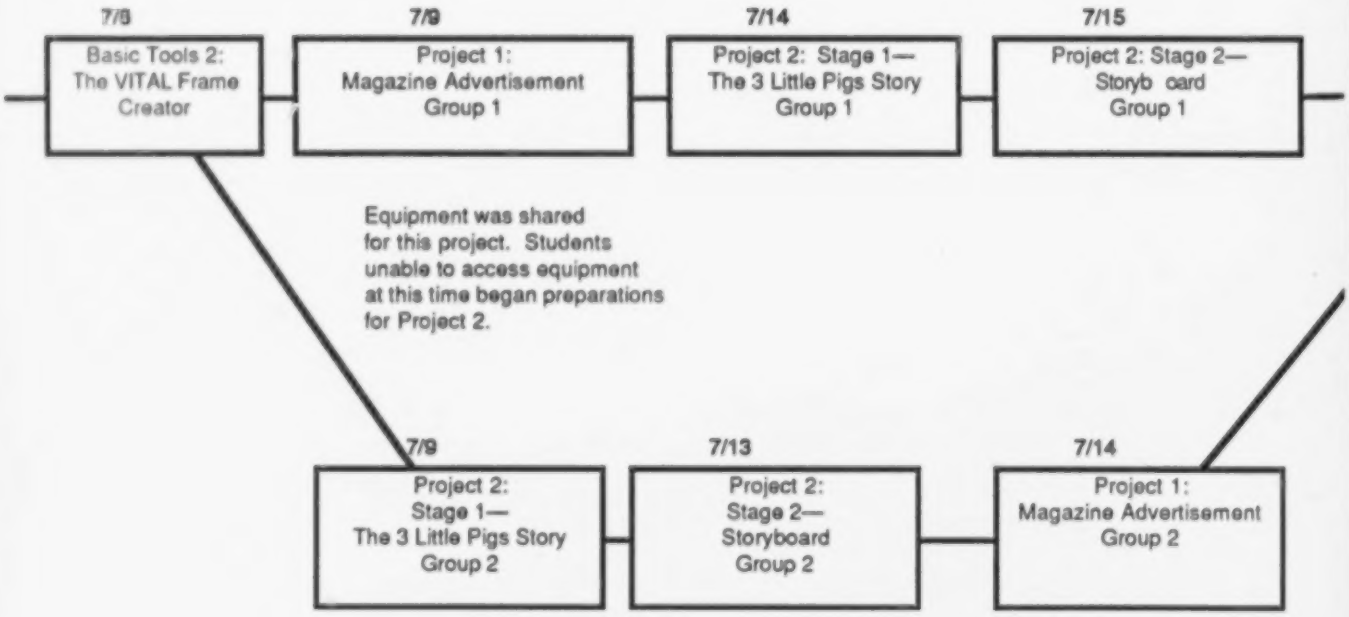
- Hypertext and Hypermedia
- Instructional Design
- Computer Graphics
- Information Technology

Tools:
Several sets of high- and low-technology tools are used:

- IBM-PC's running the course authoring system VITAL
- Apple Macintoshes running MacPaint and a hypermedia authoring program GUIDE.
- Storyboards
- Flowcharts

Project 1:
The class was split into two groups. While Group 2 works on another project, Group 1 began to 'convert' a magazine advertisement using the VITAL NAPLPS frame creator. Each student worked individually.

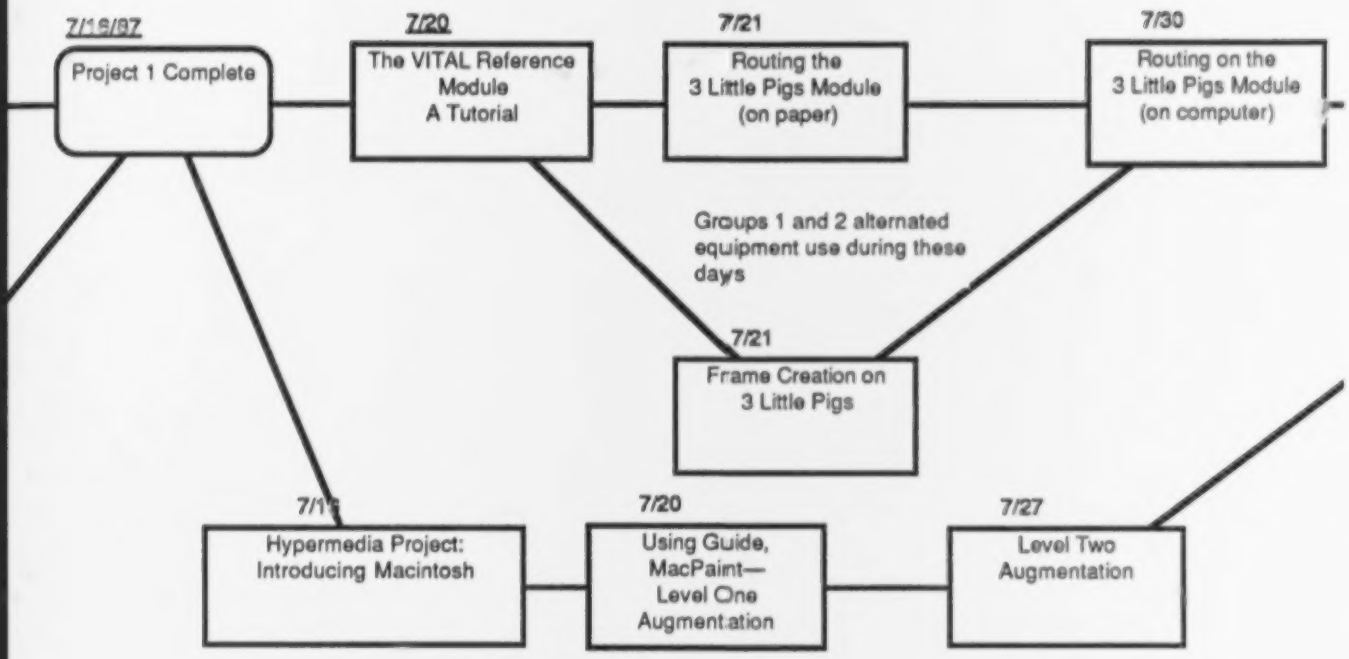
Project 2:
A group project: To create an interactive version of The Three Little Pigs.
Stage 1: Devise three different versions.
Stage 2: Storyboard each version.
Stage 3: Devise module routing for each page.
Stage 4: Implement in VITAL.


7/8
Basic Tools 2: The VITAL Frame Creator
7/9
Project 1: Magazine Advertisement Group 1
7/14
Project 2: Stage 1— The 3 Little Pigs Story Group 1
7/15
Project 2: Stage 2— Storyb oard Group 1
Equipment was shared for this project. Students unable to access equipment at this time began preparations for Project 2.
7/9
Project 2: Stage 1— The 3 Little Pigs Story Group 2
7/13
Project 2: Stage 2— Storyboard Group 2
7/14
Project 1: Magazine Advertisement Group 2

HyperMedia and Hypertext:
Hypertext is a term used to describe the ability to "read between" lines of text. A reader can select a particular word of interest and (as long as the author has provided for it) "open" it up to reveal an explantion or a clarification. Within this clarification window, there may be other words which have this ability. A reader can thus begin reading text on "electricity" and, through hypertext, end the session on a related subject by following "threads" through an electronic database.

Hypermedia is an expansion of this concept to include pictures as well as text. GUIDE, a program for the Apple Macintosh, is used to create a class hypermedia work.

Hypermedia Project:
The instructor set up a short page of text in poetic form using the program called GUIDE for the Macintosh. Each student in the class was required to select a word or phrase and 'augment' it, using more text or a picture (or both). This augmentation was identified by the program as an italisized or highlighted word.
Work continued throughout the course to develop 'threads' of associations within the piece.

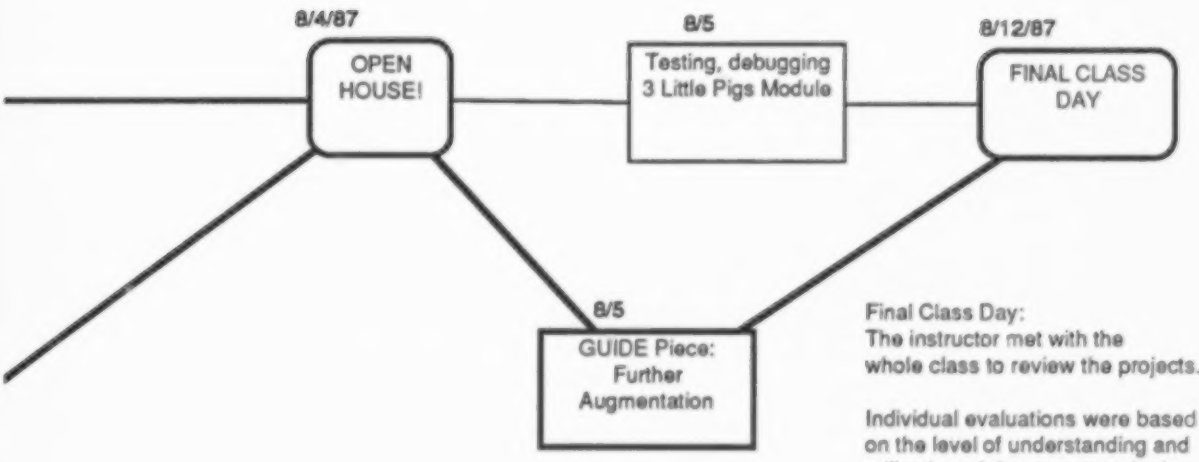


Instructional Design:
This is the term used to describe the process by which an information provider sets down goals, objectives and techniques for a project.

Instructional Design is used by educators, screenwriters and artists, albeit in different forms. For this class it is a formalization of a procedure with which most fine art students are already familiar.

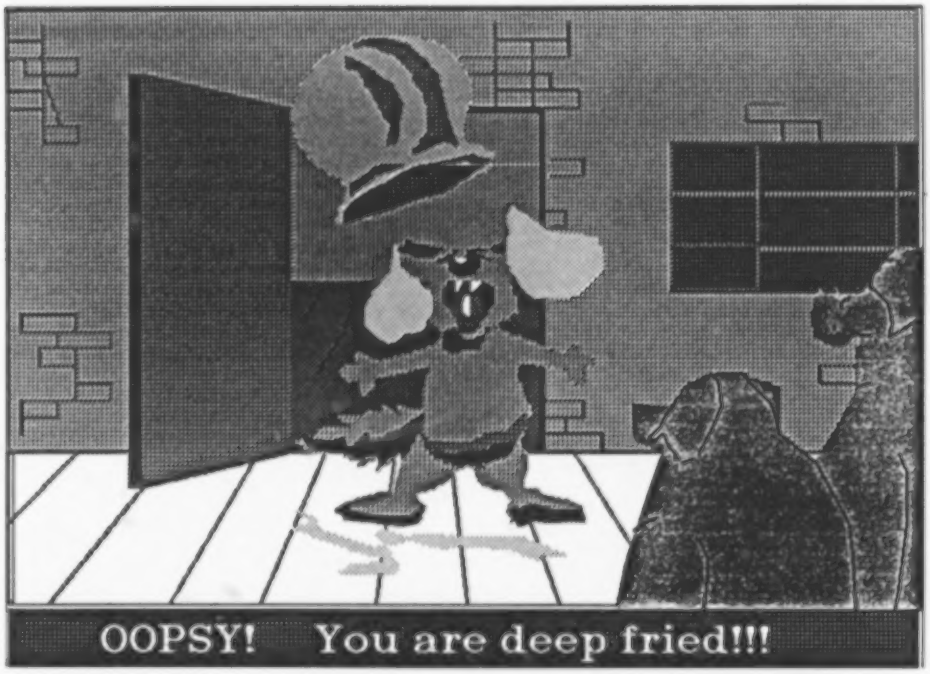
Open House, August 5:
This time was set aside to allow the students to show the University Community and other interested parties the work they've done so far, and for the instructor to evaluate current state of projects.

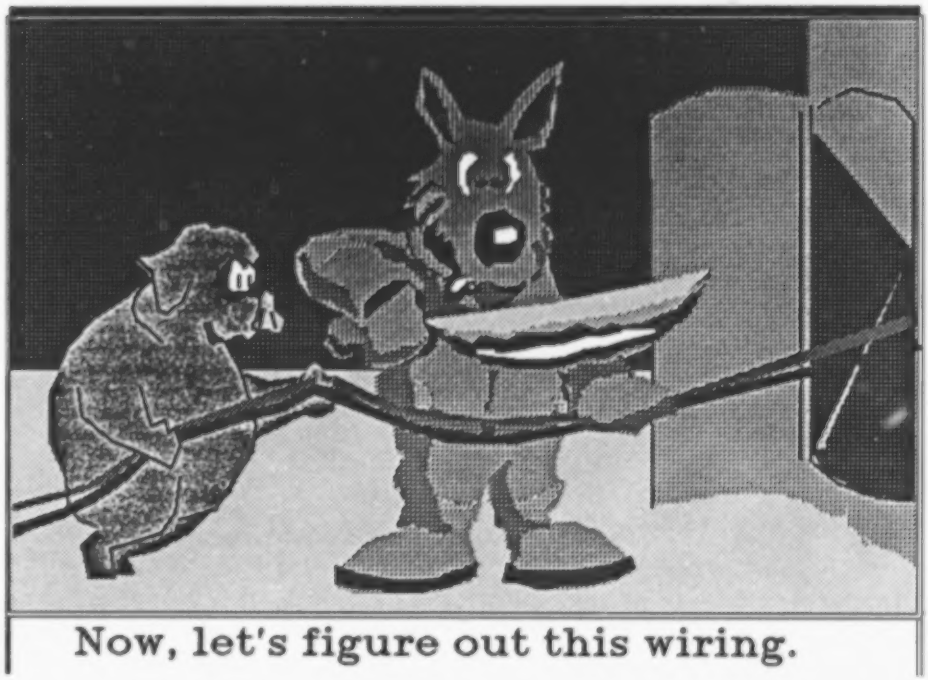
Project 3:
Students work individually on creating a small VITAL module which expands on the concept of "love".



Final Class Day:
The instructor met with the whole class to review the projects.

Individual evaluations were based on the level of understanding and utilization of the concepts dealt with in the course.


OOPSY! You are deep fried!!!


Now, let's figure out this wiring.

GEO-VITAL: A COMPUTER ENHANCED TEACHING IN GEOLOGY

I. Peter Martini and Steve Sadura
Department of Land Resource Science
University of Guelph, Guelph, Ont., Canada. N1G 2W1

OBJECTIVE

The objective of this project was to generate a set of VITAL (VIDEOTEX) modules about geology, and make them available to be accessed by students at their convenience from as many stations (terminals) as possible. More than 10 stations are available throughout the University of Guelph,nine of them in the main library. Some students have adjusted their home computer to capture the modules through a modem. The information is transmitted from a centrally located IBM PC and clones equipped with hard disk and backup facilities.

REASONS FOR THE PROJECT

The project was started because at Guelph the lectures of the introductory geology course are offered simultaneously to two classes : one, Principles of Geology (46-100), has a laboratory and is designed for science students, and the second, Study of the Earth (46-104), has no laboratory and is primarily for arts students, although several science students take it as well. The combined enrolment of the two classes in the winter semester is about 160 students. The students have a varied background, making it difficult to 'reach' each person through the classical lecture approach. Tutoring by teaching assistants is not possible because of limited manpower and funds. Computer and television techniques allow students to learn in a stimulating way at their own pace.

METHODS

Two electronic techniques, VITAL and CoSy, were chosen and implemented primarily for introductory geology, and for review material for two senior courses, Glacial Geology (46-205) and Sedimentology (46-409)- Both techniques are being perfected at the University of Guelph. VITAL is a minicomputer-based, menu-driven, educational system which derives from the televisive TELIDON (VIDEOTEX, NAPALPS codes). CoSy is a teleconferencing and electronic mailing system. Neither one is closely tied to specific - hardware at the receiving end, although a color screen is recommended for best use of VITAL.

VITAL allows information to be presented in graphic and text form, in full colors (everything is color coded), and with simple animation which can synthesize the main processes of sciences and

arts (Fig. 1). Each "screen" (page) can be prepared by anyone with less than one hour training.

However, simplification of complex concepts is best achieved through close collaboration between the instructor and talented designers. The generation of good graphics is the most time consuming and difficult part of the process. Once the graphics are generated, they can be used in various ways, in one or more courses, and rearranged to fit isolated lectures and seminars.

TEACHING APPROACHES

VITAL (Versatile Interactive System for Teaching and Learning)
The subject matter of introductory geology has been subdivided into 12 topics, essentially one topic for each semester week (Table 1). For each topic about 60-80 graphics have been generated. Several graphics are modifications of classical ones, well proven in previous teachings of the subject, however, most others were newly created.

TABLE 1. VITAL modules for Introductory Geology

Topic	Reference	Self-Test	Quiz
ORIGIN OF EARTH	x	x	
EARTH'S STRUCTURE	x	x	1. Earth str. & origin
MINERALOGY	x	x	
IGNEOUS GEOLOGY	x	x	2. Miner. & Ign. geol.
WEATHERING	x	x	
GLACIAL GEOLOGY	x	x	3. Weath.,Glac. & Sed.
SEDIMENTOLOGY	x	x	
PALEONTOLOGY	x	x	
STRATIGRAPHY	x	x	4. Paleo. & Strat.
STRUCTURAL GEOLOGY	x	x	
METAMORPHISM	x	x	5. Struct., Metam. & N. America
HISTORY OF N. AMERICA	x	x	

The graphics were presented to students in four modes:

1. Reference modules, with extensive captions
2. Self-test modules, multiple choice questions and problem solving.
3. Quizzes (4 per semester), which are marked automatically by VITAL
4. Lecture slides (or videoscope and videotape projections) with simple captions

REFERENCE MODULES. The graphics are fully labelled and captioned. The important concepts, ideas, and processes are capsulized on a screen which develops in a few seconds, but slowly enough for students to conceptually follow what is going on (the speed can be regulated from 300 to 9400 baud: at the moment we use 1200 baud). The information can be presented on a full screen, or split screen methods can be used whereby the students can select, compare and contrast things of their own choice from within a module, for instance various types of unconformities, evolutionary stages of organisms.

The challenge for the instructor and designers has been to convey information and, more importantly, to trigger specific thoughts in the minds of the students with a combination of words, forms, color codes and stylized animation. When well done, students are very responsive and opt for long sessions at each sitting (1/2 to 1 hour for each complete module).

The reference and self-test modules are made available to students as the subject matter is first presented in class, and remain available throughout the semester.

SELF-TEST MODULES. In this mode, the graphics used in the reference module, plus other very simple ones, are presented stripped of their labels, together with 'multiple choice. or 'true and false' questions, or with simple problems for the students to solve. The VITAL technique allows any type of question to be formulated and any type of response to be register, including.phrase-responses. Concatenated questions can be prepared such that various students can be led to the correct answer if they make mistakes.

Mastery of a subject can be requested before the student can progress to other subjects. When the wrong response is given to a question the computer can provide help, suggestions or "leg-pulling". We have found that, occasional outrageous, albeit polite, statements and some humour, break any tension and make students interested enough to purposely choose the wrong answer to see the reaction of the computer (that is, of the instructor that put the material in).

The main difficulty for the instructor has been in preparing good questions, subject to the space limit of a 16x40 screen. The help of good published Instructors Guides is acknowledged. However the pool of questions must be adapted to specific courses and updated, a simple thing to do with VITAL.

LECTURE MODE. Few graphics, generally without legends and with simplified captions, are transformed into "moving slides" projected with a "videoscope". This technique makes lectures professionally perfect without the need of a professional designer. We have found through experience and reaction of students, that after the firs few weeks, VITAL acquires the role of basic information dissemination and lectures can be readily used to explore in some depth basic concepts of geology.

QUIZZES. Quizzes are designed to test the progress of the students and to entice them to acquire regular work habits. This is particularly important in a semester system where proper time scheduling is the key to success.

Each quiz consists of 10 questions randomly selected from a large (200-300) pool. Each pool consists of the questions of the self test modules the students are tested on, plus other (about 50) new questions on the subject matter, and a number of questions (about 10) derived randomly from the pool of each previous quiz.

Each quiz is available for one specified week during which the students can take it at any convenient time from any available terminal, on a honor system. The fact that students may wish to take and discuss quizzes with a friend is not discouraged. We have found that each student takes the driver seat when his/her ID number and marks are on the line !

Quizzes can be administered in any way one wishes. We have selected to make VITAL (and CoSy) optional: students may gain by these techniques, including marks, but they do not loose anything if they do not use them. That is, students can try twice to achieve at least 80% in each quiz. If 80 or 90 % is achieved, the student is awarded 4 marks, if 100 % is achieved 5 marks are given. If a student does not make the 80% level, or the quiz is not tried at all, he/she does not receive anything, but nothing is lost either: the value of marks not-gained in the quizzes are added to the value of the final examination. The midterm, laboratory and final examinations are administered in the conventional, hard copy way, under surveillance.

Students have responded favourably to the VITAL-quizzes. The tension of 'test' is greatly reduced: student can try as many time they want the self-tests modules until they feel comfortable and try the quiz, with two permissible trials and not-loosing anything, not even self-respect, if they fail to make an 'A' grade. The

teacher likes a few quizzes too because for a potential total cost of 20-25 marks the following is achieved:

(1) students review the VITAL modules several times (both reference and self-test).

(2) some students who do not make the 80% score in a quiz in the first two trials, keep trying, up to 4-5 times, although they know that such extra scores do not count. These people are those that need the system most, and in trying to "beat the machine " they get a good dose of it.

(3) marking is done by computer, with enormous saving of manpower and some beneficial psychological effects (some negative too..!). One positive effect is that most students regard the instructor not as the judge, but the teacher who can provide help to "beat the computer". This is a great advantage for teaching responsive students, particularly if they have selected to use CoSy, thus can be easily reached.

CoSy (Computer Conferencing System)
VITAL can be, and is in some other courses, used alone. However, we have chosen to use it in conjunction with CoSy to provide electronic communication between students and instructors and among students themselves. Usually students prefer to use the personal 'mail' system to instructors. Only later in the semester they do adventure into expressing their opinions in 'open conferences' where everybody can read and comment. A message, mail or conference type, receives an answer within 12 to 24 hours. Students or instructors send a message from a University computer terminal, or home computer equipped with modem, to personal electronic 'mailboxes' (central University computer) or to 'baskets' of conferences. Whenever one signs on the system, he/she is informed that new mail or conference messages are available. Answers can be quickly typed and sent back. The students contribute as much as they gain from this system. They keep the instructors informed on problems, and they suggest modifications to be made to parts of the course. CoSy allows 'communication' and concise discussion: it is "academia" at its best.

CoSy is more difficult to use than VITAL. It also requires to have "something" to communicate. Every person has turned a television set on, thus can use VITAL. CoSy needs some typing skills, adherence to certain very simple but yet unforgiving rules, and many students still have a "fear of computers" . Whereas 99% of the winters '86 and '87 geology class has used VITAL consistently with many repeats of the modules, and up to just few hours before the final examination, only 70 % of the class has signed on to CoSy, and only 25% of these were very active.

EVALUATION OF THE SYSTEM

The combined VITAL-CoSy teaching enhancement system in its present configuration is best suited for local or distant education of Natural Sciences or similar. In future it will allow linkage (windowing) with minicomputer simulation programs, a 'must' for its use in senior classes and mathematical sciences, but 'not advisable' in introductory courses like geology until more students entering University will be computer-literate.

VITAL is also an electronic means of obtaining ongoing information on students progress and difficulties, tests, evaluation of specific portions of the modules and of specific self-test and quiz questions.

Students have responded in various ways to GEO-VITAL, according to their ability and learning disposition. Everybody has used VITAL extensively; most 'A' and 'high-B' students have used extensively both VITAL and CoSy, together with high attendance rate to lectures and readings of textbooks. Some science students have used CoSy primarily to obtain information and answers for laboratory exercises. Some arts students have successfully used VITAL as the primary source of information, attending very few classes.

Few students have dropped the course after the first week of lectures. The average mark for the class was in the mid-high sixties, two out of 300 students failed.The response of the students pooled during the normal evaluation of the course, has been overwhelming favourable, and essentially all of them have recommended continuation of the VITAL-CoSy system in future years.

The system may be time consuming for the instructors because of the possibility of helping the slower students through VITAL, and for being in frequent communication with the more inquisitive students via CoSy. One advantage of the electronic system is that it drastically reduces the need of office appointments for discussing problems and giving clarifications to individual students. The system is not necessarily impersonal, indeed the opposite occurs and strong electronic and class-room ties develop. The students also feel that the instructors are never farther than the closest keyboard, sometimes the one at home, without the frustration of trying to find them in the office.

VITAL, alone or preferably tied with CoSy, has great potential for distant education courses as well. The central distributing system can be reached electronically via modems, or each module can be loaded in one or two diskettes, mailed, and run in any minicomputer equipped with a color (for best results) card and monitor.

COST AND USE OF THE SYSTEM

What about cost and time involved in setting up a course like GEO-VITAL ? Some colleagues have used VITAL and/or CoSy only for parts

(few modules) of their courses, thus starting in a non-expensive way and adding as required. Even for a complete course like GEO-VITAL, a relatively small initial financial and time investment is required if the Guelph material is used , and if some minicomputer equipment is available. To start computerization of a full course from scratch, a considerable amount of money and time is required for instructors, designers and computer technical people. Such time and costs may be comparable to preparing a textbook, or double the effort needed to set up a new course, considering that much material is not only newly created, but also designed in a professional manner. However, once the bulk of the material is in, the instructors themselves or the teaching assistants, without any particular computer expertise, can easily review, re-arrange and modify graphics and captions, generate new text and graphic pages, add or remove self-test questions, quizzes, and so on: everything is menu-driven, simply and quickly. The material can be readily adjusted to the particular needs of a course given at a University or through distance education and in High Schools, and can be shared electronically with colleagues throughout the world.

ACKNOWLEDGEMENTS

Departments and people of the University of Guelph directly involved in this project were:

Department of Land Resource Science:
Peter Martini : Professor of Geology
Steve Sadura : Geological technical assistant

UNIVERSITEL (VITAL):
Les Richards : Manager
Elisabeth Black : Database Supervisor
Doris Burnfield : Design Coordinator
Mary Nairn : Design Coordinator
Andrea Nixon : Operator
Judy Walpole : Design Coordinator

Communication Services (CoSy):
Carl Gorman : Director
Tim Colwill : Operator

Technical Operations:
George Taylor : Manager

Mark Evans and Alan Campbell of the Department of Land Resource Science, contributed in introducing parts of the system to the geology classes of winter 1986 and 1987.

The project was funded by a New Initiative Grant of the University of Guelph with matching funds from the Office of Educational

Practice of the University, and some contribution from the Department of Land Resource Science. The co-recipients of the New Initiative Grant were I.P. Martini, T. Gillespie, T. Hermmann and W. Sharkey.

Figure 1. Vital pages. A. Work station to prepare or view VITAL pages, showing a list of modules of the accessed database. B. Vital page of the Glacial Geology reference module illustrating the formation of roche moutonnee. In the real screen, the figure would be color coded and show simple explanatory movements of the glacier and rock particles. Instructions to access other frames are indicated at the bottom.

REFERENCES

Black, E., and Richards, L., 1986. Tutorial Manual for VITAL. UNIVERSITEL, Office for Educational Practice, University of Guelph, Guelph, Ont., Canada, 90 pp.

CoSy Development Group, 1983. Computer Conferencing System. User's Guide Version 3.1. Communication Services. University of Guelph, Guelph, Ont., Canada. 59 pp.

Tayson Information Technology Inc., 1985. VITAL: Versatile Integrated Teaching and Learning. Operation Manual, Version 4. Tayson Information Technology Inc., Toronto, Ont., Canada.

INTENTIONAL SOCIALIZATION AND COMPUTER-MEDIATED COMMUNICATION

Dr. Elaine K. McCreary, University of Guelph
Mr. Jonn Ord, Client Services, T.V. Ontario

After decades of experience with technology transfer in the arena of agriculture the University of Guelph with its new emphasis on communication technology has discovered that many of the insights related to adoption-diffusion of a new technology through a population can be applied to computer-mediated communication (CMC) in large scale organizations. This paper describes the dissemination of CMC technology throughout one large scale organization, the Educational Data Network of Ontario (EDAN), and the special value-added aspect of socialization that enhanced simple technical competence with this new communication technology. The paper discusses reasons for the choice of distance mode for initiation of users. It presents the criteria by which sites were chosen for participation, guidelines used for design of the province-wide training program. It reports successful features of the on-line modules, and concludes with some final insights on orientation programs of this scale.

ADOPTION-DIFFUSION OF CMC THROUGH LARGE SCALE ORGANIZATIONS

Large scale organizations that have complex internal structures involving units of people who work in isolation from each other and especially when those units are geographically dispersed find that any new procedures pose special problems of introduction, orientation, initiation and instruction. When those new procedures are for communication among personnel rather than production of goods or other functions, that new mode of communication is not much less traumatic than moving the entire company into another climate and culture or even moving them into space. The impact will be significant. They will not only reproduce their former lines of communication using the new medium but they will find new functions and new lines of communication opening up.

One such experimental endeavour was undertaken by EDAN (Educational Data Network), a project jointly involving the Ontario Ministry of Education, TVOntario, and a private corporation, with consultants from the University of Guelph. The Ministry designated 75 sites for initial activity in the network, comprising participants drawn from educational businesses, universities, and primary and secondary schools throughout Ontario.

CHOICE OF DISTANCE MODE FOR INITIATION OF USERS

In its first four years of promoting CMC throughout a community of 770 faculty, 2500 fulltime support staff, and 12,700 degree program students, the University of Guelph came to conclude that the most effective means of initiating new users was through a person-to-person intensive training program of an hour or two, involving closely guided hands-on practice with the equipment and software, heavily supported with a user manual and most importantly followed up with some tutoring involving a "buddy". The buddy system of immediate friendly feedback, advice and user support seemed in our experience to be by far the most significant variable. However, for the Educational Data Access Network of Ontario, neither of the two most important features were going to be financial feasible. A face-to-face initial training that would require the transportation and housing of 75 participants to a central site or even several regional sites was out of the question and one of the key features of their activity in their home bases was that there were no colleagues equally, let alone better, prepared in technical matters involving computer-mediated communication.

Add to the restrictions of financial parameters time constraints where an experimental program is given three to five months to prove its usefulness, then a second set of problems emerges. Experience has shown that a novice user with untested equipment and a manual, in three months is barely literate enough to begin to make use of electronic mail and begin to see the possibilities of the system. Far more than this was required for a successful test period within the three-month time horizon.

However, despite financial constraints, the Ministry of Education and T.V. Ontario are both committed to equalizing opportunity for learning throughout all regions of Ontario, and therefore decided to make participation in EDAN available to any site that could conform to the following criteria. The host site had to be able to provide hardware in the form of a microcomputer and modem. It had to have phone line access. It had to provide one of its personnel as a demonstrably interested participant who would make a time commitment of a total of ten (10) hours distributed as 1/2 hour sessions per working day, five days a week for two weeks. Half an hour per day was a manageable time commitment for participants to make, and it influenced us as designers not only to opt for an initiation process that would be principally distance mode, but to emphasize on-line training rather than paper instructional modules. This, we felt, would maximize the trainees' hands-on engagement with the technology and minimize disruption to their regular work routines.

GUIDELINES FOR DESIGN OF THE PROVINCE-WIDE TRAINING PROGRAM

Goals

The training program was structured to give advance on-line instruction over ten half-hour periods lasting two weeks. Participants would then come together in regional workshops for special follow-up training with workshop specialists from TVOntario. The purpose of the on-line training in advance of face-to-face meetings was twofold. Firstly, it had to be written in such a way as to enable 75 isolated new users to develop skills in a self-conducted manner with the software of COSY--the conferencing system that they were learning.

Secondly, the on-line training was designed to give them guided interactive experiences with other users on the assumption that their ability to interact, discuss, negotiate, argue, reconcile, and reach decisions with colleagues was the real purpose of the training not simply keyboard competence to write electronic memos.

Style

With the two goals in mind, an intensive compact skill development and interactive socialization to this mode of communication, several points of style were decided upon. Firstly, that the writing instructional style, like most distance education, would be chatty and upbeat for the purpose of encouraging learners who were otherwise working in social isolation. Secondly, that there would be a carefully sequenced set of programmed tasks with short-term reinforcement to encourage the new user with a sense of progress, security, and increasing confidence.

User Supports

Like most distance education systems, special features of user support had to be added to this province-wide training program. Principally these were administrative and technical support. Correspondence was undertaken in advance with administrators at each of the 75 sites to solicit their participation and cooperation, and to indentify the local staff member who would be suitable as the first to learn this new system. Telephone contact had to augment other means of correspondence. Once people received the small-scale paper module that accompanied the on-line training, their first problem was to make that initial electronic contact with the host computer at TVOntario. Here again a telephone hotline was essential recourse for isolated users who had no technical advisors on site. Once on-line, there was a detailed "Help Hierarchy" of options to refer to including programmed help menus and access to experts in an on-line conference.

Workshops

In order to accommodate a constituency of 75 learners with only

three central staff, training packages were mailed out to two sub-sets or cohorts of learners per week. As their two week on-line training was completed, a regional workshop would be convened -- principally to consider what their professional applications of this now familiar medium might be and to help them move directly into setting up and moderating appropriate conferences. Timing was arranged so that as each set of workshops was completed, another cohort of on-line learners would be coming due for their follow-up workshops.

SUCESSFUL FEATURES OF THE ON-LINE MODULES

Standard Module Format

The first feature of the ten on-line modules that worked very well was its standard format. Each day had a "to begin" section that lead the user to the on-line location for that day. Each module had "today's commands", a list of the new commands that would be learned, and the full set that would be used on that day. Each day had "supplementary instructions" if any were necessary to carry out the day's task. Each day indicated" electronic work spaces" that would be in use helping the new user to think in terms of discrete areas of work within the system.

Sequence of Modules

Each additional unit of commands was sequenced in an order compatible with new user interest: writing options before editing skills, read options before writing, getting on and off the system successfully before any other task at all. Almost all frequently used commands were mastered in the first five modules. Module 6 required they work with others in triad in "conversation mode". Module 7 was a tour de force review of previous commands. Module 8 taught users to move from mode to mode acquiring a private database using the "file" command. In module 9, they learned to print paper copy of their private and team creations. Module 10 moved the whole consituency of new users to a discussion of applications in preparation for their upcoming workshop.

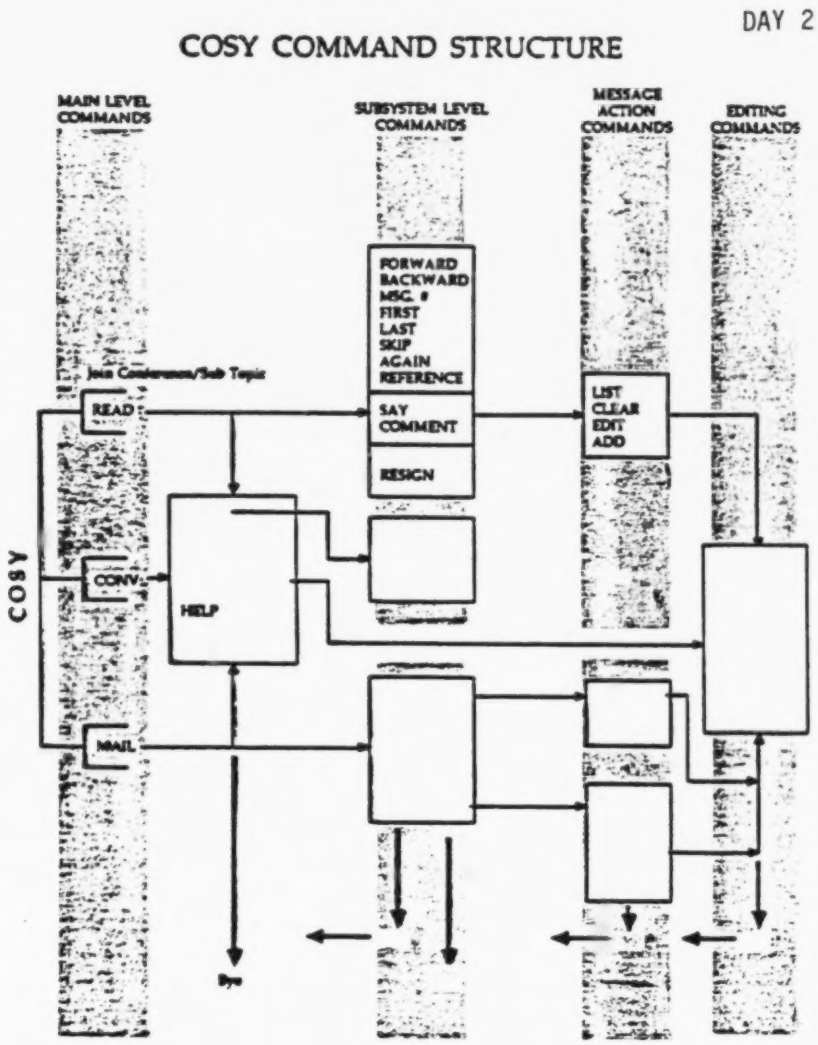
Screen:Paper Ratio

The whole intention of the on-line training was for the learner to experience continuous response and coaching from the training writers as driven by the learner's own actions and prompts. Given an instruction, the new user would try something and either be assured it worked, or be given a corrective procedure on-line, or through the safety feature of the paper back-up kit. Paper was essential at the first sign-on and that format for opening the day's work was kept just to increase the novice user's sense of security. Throughout, we tried to proportion on-screen activities to paper back-up at about 80:20.

Progress Maps of the Command Structure

A feature that worked particularly well for visual learners were the maps. These were graphic illustrations of the training sequence show as full command structure through its four levels, with the help section accessible from any of the other levels.

PROGRESS MAP OF DAY 2

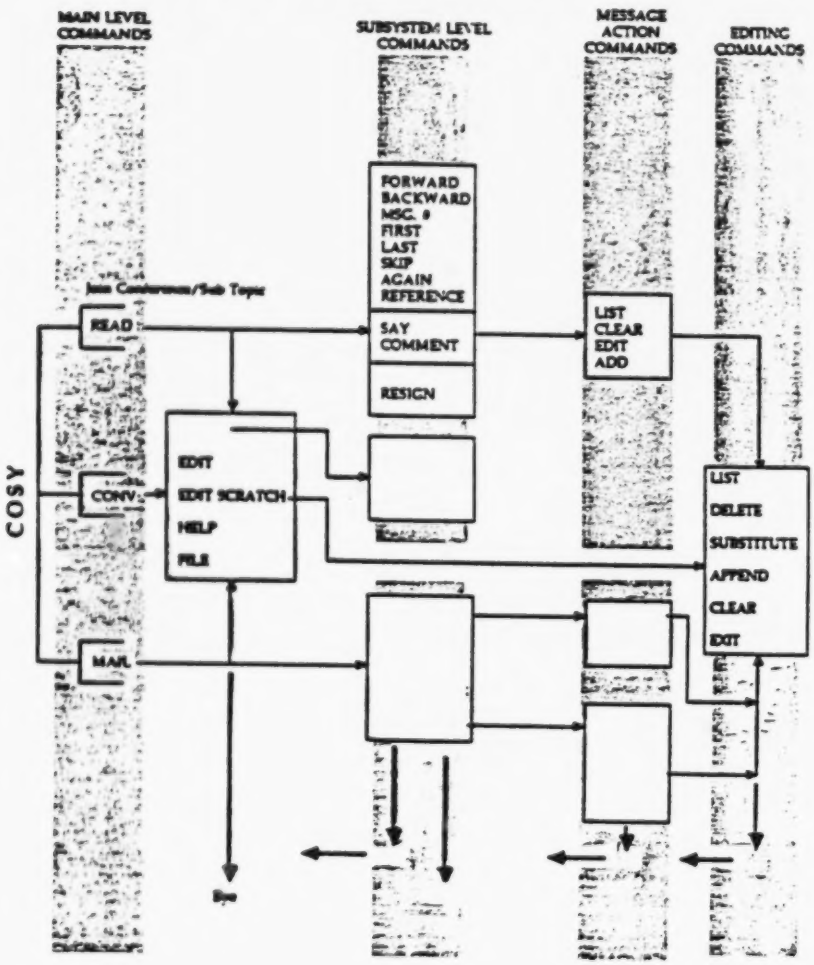


This overview of the COSY command structure was first shown as a set of interlocking blanks. As they proceeded into a new sub-set of commands each day, those would be shown in their appropriate place within the graphic of the COSY command structure. In this way, users were given an advance organizer that became a more explicitly articulated road map of the whole command structure as they personally added new sets of competencies.

PROGRESS MAP OF DAY 3

DAY 3

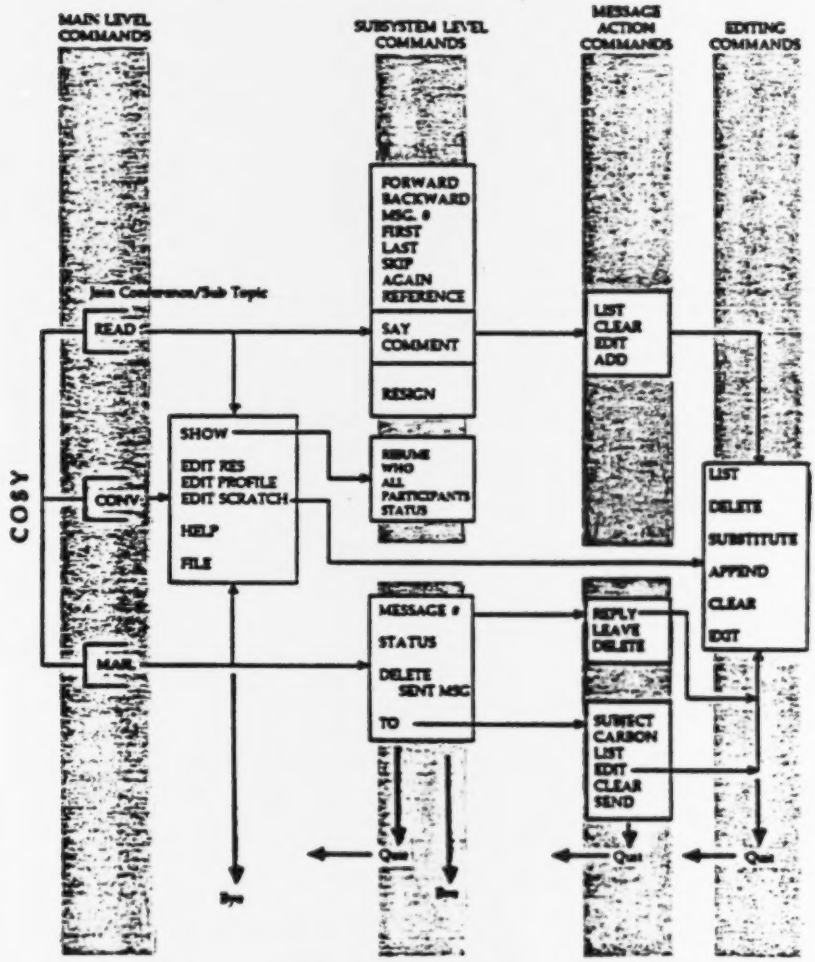
COSY COMMAND STRUCTURE



PROGRESS MAP OF DAY 5

DAY 5

COSY COMMAND STRUCTURE



SOCIALIZATION SEQUENCE

From the beginning, new users were moved through a set of small and large group tasks requiring them to engage in some of the various social environments that CMC provides.

As soon as they could write, even before they could make editorial corrections in a message, they were required to send a message to their tutor or moderator, to indicate that they had made it that far through the modules. The purpose was to ensure that they would receive personalized mail very early on when the tutor responded to their check-in.

When they had learned editing skills, they were set a casual task of reading and making small talk in a social mixer called "Fireside", while elsewhere they worked on the more onerous task of editing a personal resume for posting where all the community of new users would be able to read it. "Fireside" proved a social vehicle for some people dispersed throughout the province who had in fact met face-to-face before at professional conferences.

Their chatting was visible to newcomers, providing some role modeling that newcomers could consider in their "required" conversational triad with two far-distance buddies whom they had not met before. The triad had a task to complete, in the form of a brainstorm on "sources of stress in the workplace". Dealing with serious enough issues, it was nevertheless intended to be an amusing exercise...out of which the triad would do some decision making and present to the whole community a short list of irritants and antidotes for sources of stress in the workplace. In the last module, people began to pool their suggestions for the workshop to follow that would address professional applications of CMC province-wide, principally for methods and curriculum working groups.

In the meantime, more than 20 special interest conversations had sprung up, completing the possible combinations of small group/large group and task-oriented/free flow discussions, as illustrated in the figure below.

Intentional Socialization Experiences

	Free-Flow Discussions	Task-Oriented Discussions
Large Group	"FIRESIDE" social mixer	Professional application to methods & curriculum
Small Group	20+ Special Interest Conversations	"Less.Stress" conference

USER SUPPORT DURING LARGE-SCALE ORIENTATION

Care was taken to foster the process of change required of the new users by establishing consistent contact, support and direction during implementation of the training sequence. This process began during the identification of sites by EDAN staff; it increased during on-line training with the introduction of 'on-line buddies' or working triads of learning partners that contributed to a sense of mutual support and team obligation. the sense of group was extended to the larger community through the face-to-face workshop with an immediate invitation by each special interest group to the rest of the community to participate in a new conference addressing their professional interests.

One former user of the system who had just participated in this training commented that "the whole system seemed much more user friendly". The user mistakenly assumed that the conferencing software had been greatly re-written to achieve this result.

Consistent and persistent pre- and post-training contact was essential to a successful experience for isolated new users. A certain amount of creative and supportive pressure was useful to ensure that the new user not only complied but initiated new behaviour, continued and was reinforced in that new behaviour by positive results. Throughout the experience it was vital for participants to feel sustained support from peer groups, administration and project personnel to ensure that the initial pressure and effort did not result in discouragement and alientation.

FINAL THOUGHTS

Socialization aspects of CMC communities in the workplace and elsewhere are going to be posing us interesting problems for years to come. It was our experience that the additional effort required to build in a range of social experiences, in order to enable new users of the medium to become accustomed to the idioms, the pacing, the courtesy, and the affective aspects of on-line communities was well worth the effort. Future efforts can only ensure that we become more refined in our identification and encouragement of behaviours that will make on-line communities civilized places to live, and productive places to work.

ISSUES IN THE DEVELOPMENT OF INTELLIGENT TUTORING SYSTEMS

Mary McLeish and Karl Langton
Department of Computing
& Information Science
University of Guelph
Guelph, Ontario

May 1987

Abstract

Intelligent tutoring systems (sometimes known as Intelligent Computer Aided Instruction, ICAI) have grown out of adaptive instructional systems. Such systems try to combine problem solving experience and 'discovery' learning with the effective guidance of tutorial interactions. It becomes important to develop tutorial strategies which know "when" to interrupt a student's activity, "what" to say and "how" best to say it.

The process of developing a system with such capabilities begins with the problem of forming an adequate model of student behavior. One project at Guelph involves designing and building an Intelligent Tutoring System for the Artificial Intelligence Language Prolog. In order to construct a student model for the system, experiments are being carried out to study student's programs and thought processes in a variety of ways. "Buggy" programs are captured and their study is contrasted with the results of questionnaires. The relevance of prior knowledge of other programming languages to the student's learning capabilities is also investigated.

This paper highlights features of the proposed system that go beyond CAI and discusses the work already done at Guelph on student (user) modeling issues in the context of learning Prolog.

Introduction

Studies of students learning cognitive skills have shown that private tutoring seems to be much more effective than conventional classroom instruction. In the classroom, students have to listen to lectures, read texts, and then later work alone on homework problems. Private tutoring on the other hand, provides the student with an experienced person to guide the reading and problem solving. This type of one-on-one activity paves the way for a more intimate interaction.

The notion of providing students with individual tutors has in the past been an unrealistic prospect for teaching institutions.

Staff time and cost considerations have been the obstacles. However, with recent developments in computer technology, individual tutoring is now both feasible and relatively inexpensive. In fact, computer assisted instruction (CAI) has been a part of many school curricula for the past decade.

With the recent advances in the area of computer science known as artificial intelligence (AI), methods have been developed which can be used to enhance CAI. The outcome of this has come to be known as intelligent tutoring (IT) or intelligent computer assisted instruction (ICAI).

Intelligent tutoring differs from other CAI approaches to education by trying to provide effective instruction to problem solving the same way that a human tutor does. A number of the more traditional CAI systems were "automatic page turning" or "fill in the blank" instruction programs. While these programs are indeed very useful, they often assume all students to be equal.

Another trend in computer assisted instruction programs has been the development of programs to simulate laboratory equipment. The wide range of simulation programs (see proceedings of this event) seems to be in a period of explosion. For example, there is even a program available (City Software, 1985) to simulate the use of a nuclear magnetic spectrometer. High resolution graphics and faster processors have made these programs a useful alternative to the rising cost of laboratory equipment.

The development of more "personal" interactive computer programs has resulted from the realization that students do not all have the same problems and needs. Intelligent tutoring systems try to address the individual student's strengths and weaknesses; by so doing, the tutor can avoid redundant sessions and provide extra instruction on the more troublesome ones. By utilizing these concepts, programs have been developed that can effectively create customized teaching strategies.

This paper presents some of the issues concerning the development of computer tutors. We begin with a description of the various components associated with a typical intelligent tutoring system and how they fit together. The focus of the following section is on acquiring knowledge about student problem areas. This can be useful in the design of a "bug catalog". This, in turn, is crucial to the design of a student modeling component, which is then described and examples of existing systems are given. The discussions are based on experience gained in our efforts to develop an intelligent tutoring system for teaching students how to program in the AI language, Prolog. The last section describes the need for a Prolog tutor. Finally, the paper is summarized and concluded.

Components of an Intelligent Tutoring System

Intelligent tutoring systems are made up of a number of modularized components (see figure 1). The "domain expert" has background information of the specific domain of study. The "tutoring expert" is the part of the system which contains an expert system for instruction. This component has problem solving, explanation, and diagnostic modeling strategies built into it. With this information the system can present material to the student and process the student's response. In order to build a tutoring expert, a "bug catalog" containing a list of possible problem areas should be developed. This provides the developer with essential information in deciding what types of problems assistance can be given on. The tutoring expert can determine what a student knows and what s/he is confused about (i.e., what "bugs" the student has), when the student should be interrupted and what should be said, and what further sessions the student should perform. In order to provide this kind of tailored assistance, a model of the student's knowledge and difficulties must be maintained, in what is called a "student model".

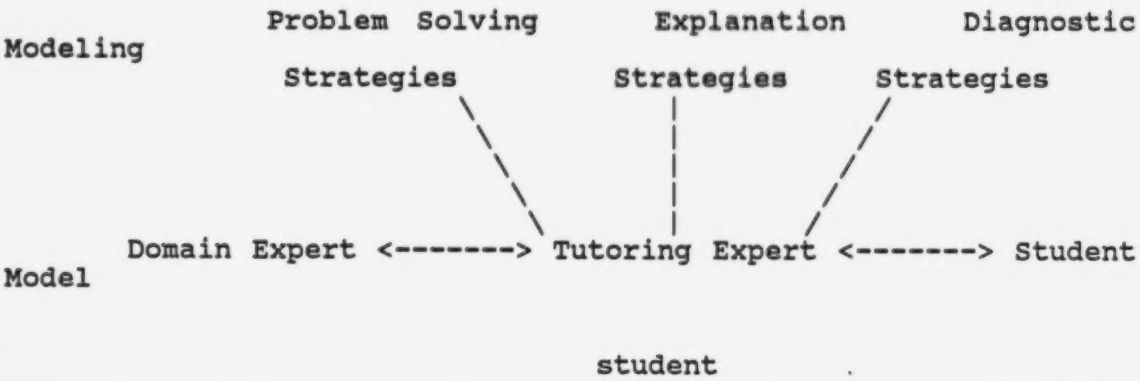


FIGURE 1: A Typical Intelligent Tutoring System

The following section examines how we went about gathering information to develop the tutoring expert, with a focus on what kinds of problems typical students would encounter.

Knowledge Acquisition and the Development of a Bug Catalogue

In order to build a domain expert, information about the particular domain must be gathered and made explicit. This

process has come to be known as "knowledge acquisition" (Hart 1986). The domain expert should have a thorough knowledge of the various aspects of the area under investigation. There are several ways of acquiring knowledge, each of which are quite unstructured. The ones examined in this section include monitoring programming sessions and handing out questionnaires.

One method of acquiring information about how students learn programming is to monitor their programming sessions. This can be done either by actually watching the students as they work, or by "grabbing" their edit sessions.

First we developed a programming problem and gave it to a group of computer science students as an assignment. As the students were working on their programs, we would "grab" all their editing sessions. By looking at each students successive attempt at a solution we hoped to be able to gain insight on how they were thinking and what kinds of problems they encountered. Upon completion of the assignment we gathered together the edit sessions, made hardcopies of them, and began the painstaking process of analysis. We found that the successive edit sessions were characterized by zero, one, two, or more changes. To account for the program versions with zero changes we speculated that the students had ended their edit sessions due to time constraints and personal reasons (e.g., tiredness, hunger, other work to do, etc.). We also found many of the problems to be due to spelling mistakes. The more program related problems included: improper use of variable names, unexpected backtracking, and recursion related issues. It did provide us with a rudimentary bug catalogue, but also led us to look for alternate methods of data collection.

At the end of the experiment the following new questions arose: How could the method of analyzing program versions be automated? Was the programming problem a relevant one to base conclusions on? Would the insight gained be useful in the development of student models?

As mentioned previously, another way to learn about student difficulties is to actually watch them program. This is useful because when students need help, they will often ask. This method of videotaping groups of students has been more widely advocated recently (Curtis 1986). Also, if you are there watching someone making changes in their program, it's easy to ask them why they made the change. With both of these situations occurring, a great deal of insight can be gleaned. We have yet to test this method, but it is fair to say that however one goes about acquiring knowledge of student difficulties, it is an incremental and rather unexplored area; there are no formal methods or general rules which can be applied.

What is a Student Model?

We have assumed that since most students have different learning rates and that the problems they encounter are often different from student to student, there is a need for more student specific tutoring strategies (Kearsley 1987). This has led to the notion of developing a "student model" to store each student's current knowledge state. This section examines the need for student models and discusses how one is developed.

The term "user model" became popular in AI during the early 1980's. A user model usually means "some form of representation of the person using the computer system". Many aspects of AI regard user models as an important tool for use in human-computer interfaces. Besides intelligent tutoring systems, other areas where user models have been employed include expert systems (Clancey 1981, Cohen 1986) and natural language systems (Carbonell 1985).

As noted by Wahlster (Wahlster and Kobsa 1986):

> "A user modeling component is that part of a dialogue system whose function is to incrementally build up a user model; to store update, and delete entries in it; to maintain the consistency of the model; and to supply other components of the system with assumptions about the user."

The incremental building of a model of the user's cognitive abilities and preferences enables a system to respond in a most appropriate way for the person using the system.

When we refer to a system that is specifically to be used as a teaching or instruction system we modify the term "user model" to become "student model". Thus, a student model is a computer system's model of the student's current state of knowledge and beliefs about a particular domain.

As noted by Van Arragon (Van Arragon 1986) early CAI systems were quite simple in that they were often:

> "drill and practice programs that would quiz a student, and then keep track of the number of right and wrong answers so that the questions can get harder or easier based on the student's progress"

Today, modern teaching facilities have replaced the ancient boarding school methods of "drill and practice" routines, with the more cognitively intuitive methods of "learning by

discovery". Along with new teaching methods in the school systems, the development of CAI programs is going through similar changes with the incorporation of student models.

The introduction of student models has enabled programs to predict students' levels of understanding and to recognize their particular learning style. Some students learn best by analogy, while others through experience, and even others by being told. A good introduction to related AI research in learning can be found in the book, "Machine Learning" (Michalski et al, 1983).

Whatever the preferred learning style, the system should be able to make inferences from students' responses to questions and exercises, modifying its teaching strategies accordingly. A student modeling component should also be able to recognize and correct student misconceptions and deviations. Therefore, a model should be flexible enough to store and retrieve a wide range of student information.

Example Intelligent Tutoring Programs with User Models

Early CAI systems had no student models. However, as CAI programs became more "intelligent" with the help of AI techniques, systems began to evolve with "overlay" and "bug" modelers. An overlay modeler uses stereotypical information about the domain to store whether topics (or skills) are known or unknown. A bug modeler works by storing information about bugs or misconceptions. These modeling constructs have been built into the two ICAI systems, West (Burton and Brown, 1982) and Buggy (Brown and Burton, 1978).

West is an ICAI program which evaluates and suggests improvements to students' learning elementary arithmetic constructs. Learning is facilitated by having the interface set up as a board game. The object is to move a player across the board by writing and solving algebraic equations which determine the distance to be moved. Whenever the student's solution is different from how the "domain expert" would solve the problem, the student is provided with hints on how to make better moves.

The program Buggy models bugs made by students solving arithmetic problems. It was designed to give teachers experience in diagnosing students' incorrect procedures. Buggy gives teachers experience in discovering the underlying causes of student errors by presenting examples of bugs. Then the teacher has to try to deduce what the problem is with the particular buggy example.

The tutor we are developing is to teach the AI language, Prolog, to computer science students as a second programming language. Computer programs that have been developed to teach programming

include Proust (Johnson and Soloway 1985), the LISP Tutor (Anderson and Reiser 1985), and Scent (McCalla et al 1985).

PROUST is a system that takes programs written in Pascal by novice programmers and performs a "bug analysis". The program not only points out mistakes to students, it also determines how the bug can be corrected and suggests why the bug arose in the first place.
The LISP Tutor is an ICAI program used to teach the AI language LISP to students. The program works by presenting a series of tutoring modules to students where they are given requests to work with and develop various aspects of LISP programming tasks. As the student is writing the solution, the tutor watches, trying to figure out what correct or buggy information could have led the current input. When the input is buggy, the tutor interrupts and gives advice. There are several modeling components in this program including an ideal student model, a buggy model, and a current student model.

SCENT is an ICAI program that also teaches students to program in LISP. Its current phase is to provide sophisticated debugging assistance to novice students through the use of extensive trace facilities. This project is still in its early phases, but its main intention has been to provide computer science students with programming environments that will induce learning. It will undoubtedly make use of expert models and student models.

Other features which can be added to ICAI programs to facilitate learning and to make them less intimidating to students are natural language understanding front-ends and the use of cooperative strategies. Natural language front-ends enable students to converse in English with the tutoring system. Cooperative strategies include such things as spelling correction and misconception correction modules. These features address the plans and goals of the student using the ICAI system.

Student Modelling in the Context of a Prolog Tutor

The project at hand attempts to advise students according to the premise that there are interference effects when learning Prolog, which come from a student's ingrained knowledge of procedural languages (like Pascal). The following example programs illustrate some of the differences between the two languages. This database program is to assist in building a genealogy to determine if various people are related to one another. While the programming would be simple in Prolog due to its built in searching technique of backtracking, the same program done in Pascal would be quite tedious. Figures 2 and 3 illustrate these programs.

```
            parent( Parent, Child )
        /* The above FACT states that the variable "Parent" */
        /* is the parent of the variable "Child")           */

            ancestor( RelativeX, RelativeY ) :-
                        parent( RelativeX, RelativeY).
            ancestor( RelativeX, RelativeY ) :-
                        parent( RelativeX, Child ),
                        ancestor( Child, RelativeY ).

/* The above RULE states that "RelativeX" is an ancestor of */
/* "RelativeY" if "RelativeX" is the parent of "RelativeY"  */
/* or if both "RelativeX" is the parent of "Child" and      */
/* "Child" is an ancestor of "RelativeY"                    */

...the database:

parent( Baden, Karl ).
parent( Geofrey, Baden ).
parent( Karl, Zoe ).
parent( Baden, Jennifer ).
parent( Jennifer, Braeda ).
parent( Norah, Karl ).
parent( Sarah, Norah ).
parent( Serina, Sarah ).
```

FIGURE 2: Prolog Database Program

In order to find out whether or not two people are related to one another, all the person using the system has to do is type in the following:

```
ancestor( RelativeX, RelativeY ).
```

(Where RelativeX and RelativeY are substituted with the names of the actual people in question.)

For example, typing in: ancestor(Baden, Zoe)
Would produce the response: yes.

```
program ancestor( input, output, relationfile );

type
     relation = record
     parent : string;
     child : string
     end;
var
     relationfile : file of relation;
     done : boolean;
     name1,
     name2 : string;
```

```
     response,
     result : char;
procedure search(       );
begin
1) check to see if there is a parent = name1
2) if there is, check to see if there is a child = name2
     if there is, then stop, the search was successful
     otherwise check to see if there is a child of a
         child (or child of a child's child, etc)
         that is = name2
3) continue until a successful search has been achieved
  or the range of possibilities has been exhausted
end;
procedure ancestortest( name1, name2 : string, result :
                                                  char );
begin
     reset( relationfile );
     while not eof( relationfile ) do begin
     search( relationfile );
          ...
     end;
{    if the search was successful then
       result = 't'
     else
     result = 'f';       }
end; { ancestortest }

begin { main }
     done = false;
     repeat
         writeln( 'enter two names: ' );
         read( name1 ); readln( name2 );
         ancestortest( name1, name2, result );
         if result = 'y' then
             writeln( 'yes, they are related' )
         else
             writeln( 'no, they are not related' );
         writeln( 'like to try another pair (y/n) ?');
         readln( response );
         if response <> 'y' then
             done = true;
     until
         done = true;
end. { main }
```

Relationfile:

Baden Karl
Geofrey Baden
Karl Zoe
Baden Jennifer
Jennifer Braeda
Norah Karl
Sarah Norah
Serina Sarah

FIGURE 3: Pascal Database Program

While the above program is not quite complete, already one can see there is a lot more programming involved to develop a database program similar to it's Prolog counterpart. While we do realize there is a lot of input/output and search coding for the Pascal program, this is exactly the point. Prolog has a built in search facility, and questions can be interactively posed to the database without a need to do any input/output programming. For similar database and knowledge base development applications, Prolog is by far superior than Pascal. This has been a reason why it has been chosen by many AI researchers to be the language of choice for the development of expert systems. We feel that since there has been such a thrust to learn to program in Prolog, there is a need to have sophisticated intelligent tutoring programs to assist students learning the language. However, because the language is relatively new, most students are learning it after significant exposure to other 'Pascal like' languages. Some major language differences are listed below.

in Prolog:

1) There is no looping facility other than recursion.
2) There are no assignment statements.
3) There is a built in search facility (depth first).
4) The use of list structures is important (as in Lisp).
5) Answers are obtainable only by querying a database
6) Program efficiency can be obtained through the use of a special 'cut' operator, which is tricky to use and without a procedural parallel.

(The extent of these differences can depend on the particular Prolog implementation.)

In teaching Prolog to our undergraduate classes at Guelph, instructors and T.A.'s would repeatedly hear students complain and struggle with efforts to reconvert their thinking to the mind set of declarative languages. However, a study of the 'buggy'

programs alone did not unearth enough evidence to really support this theory.

As a result, other knowledge acquisition techniques were tried and are underway. Questionnaires were designed to ask the students directly how they felt about any interference problems from other programming languages. These initial questionnaires were circulated to a 4th year/graduate class (486/686) at the University of Waterloo, where the students were learning Prolog for the first time, after many years of procedural experience. The sample size was 85. Another class (200 students) was surveyed with the same questionnaire. This class was at a second year level and the students were being taught Prolog in the same course as Pascal, but following it. A sample of the questionnaire and an analysis of some results is shown in figures 4 and 5 (see appendix). It is interesting to note that there appears to be less of an interference problem with students introduced to Prolog at an earlier stage, even if it follows Pascal. (Actually these students already had some exposure to Fortran and Basic.) Control flow and the use of recursion were two areas which seemed to cause difficulties. Further questionnaires were circulated pinpointing particular problem areas concerning a recursive routine the students had been asked to program. The questionnaires proved to be valuable, but in a different sense from the study of 'buggy' programs. A system is now being automated which attempts to be a combination of the two techniques of data collection. Every time a student edits a new version of his program, he is prompted with a set of questions asking him what he thinks was wrong with the earlier version and how he has attempted to correct the mistakes. The 'buggy' programs and editorial comments can then be studied, comparing the student's perceptions with the actual problems.

Prototype System

Several systems have already been developed primarily to debug Prolog programs (and indeed most Prolog implementations now come equipped with fairly sophisticated debugging capabilities.) The work of Shapiro (Shapiro 1982) uses a formal grammar approach, whereas some work carried out at the University of Calgary employs a graphical debugger. A recent technique employed by Hartley (Hartley 1986) compares a student trace with the 'true' program trace in order to track for errors. We are developing a system which concentrates more on a true tutoring capability, as opposed to a debugging facility. When an error is detected at run time, the system will suggest a variety of possible causes based on the study of the bug catalogue, questionnaires, and the premise that certain types of errors arise from the interference phenomena discussed earlier. The system can then provide a list of possible causes. The student can choose to request further information on one of these, giving him further suggestions of

where errors could likely occur. The student will be pushed, as much as possible, to discover the exact cause of the error himself and to correct it. As previously mentioned, some existing systems already make use of a prescreening technique to decide what sort of background a student has and then tailors the tutorial help appropriately. Questions along the line of the first question on the questionnaire of figure 4 (see appendix) could be used, for example, to guide the system in deciding on the extent to which errors can be explained by background problems.

Further refinements include prompts asking the student what he intended the outcome (final result) to be and then comparing this with the actual outcome. (This is for situations where either complete or partial run time results were obtained, but they were incorrect.) The knowledge base of the system envisioned can be updated with the information collected on students using the system.

Summary and Conclusion

This paper has discussed some of the issues associated with the development of intelligent tutoring systems. The various components associated with a typical intelligent tutoring system have been discussed with a particular emphasis on how to acquire the knowledge necessary to understand what some of the specific problem areas for the students may be.

The present status of a recently conceived project at Guelph concerning the development of a Prolog tutor has been discussed in this context. Clearly much work remains to be done. However, it is felt that a much better system will be developed if more attention is paid to studying student errors ahead of time, instead of designing a system first and then going back to study problem areas.

Appendix

AI - Prolog Feb. 1986

Phase One: Prolog Questionnaire

NOTE: The purpose of this questionnaire is to try to gain some insight into the problems associated with the learning of Prolog after one has gained some familiarity with other programming languages.

1) Please list the programming languages which you have had some experience programming with prior to learning prolog:

2) We are particularly concerned with difficulties in learning prolog arising from interference caused by previous programming experience in other languages. Please indicate the extent to which this has been a factor in your learning the prolog concepts listed below:

	Not a factor	Caused some difficulty	Caused most of the difficulty
a) Syntax			
b) Control Flow			
c) I/O			
d) List Processing			
e) Recursion			
f) Backtracking			
g) Cut			
h) Logic Operators			
i) Database Operations			
j) Complex Conditions			

FIGURE 4(1)

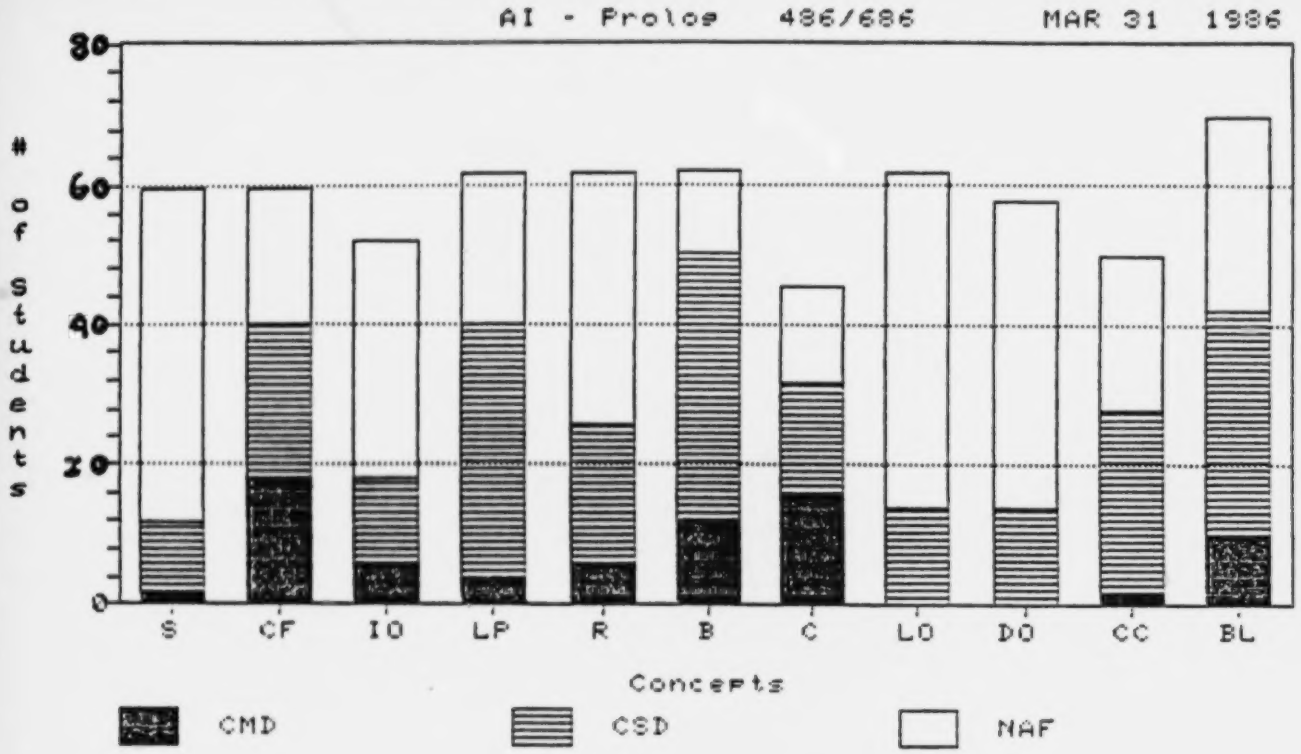
3) Finally, if you feel that there are indeed problems caused by your background in other languages when learning prolog:

do they i) subside after a few weeks into prolog ☐

ii) continue to influence your prolog programming abilities ☐

iii) only cause a problem with 'getting started' in Prolog on the first programming problem you attempt. ☐

FIGURE 4 (2)


AI - Prolog 486/686 MAR 31 1986
# of Students
80
60
40
20
0
S
CF
IO
LP
R
B
C
LO
DO
CC
BL
Concepts
CMD
CSD
NAF

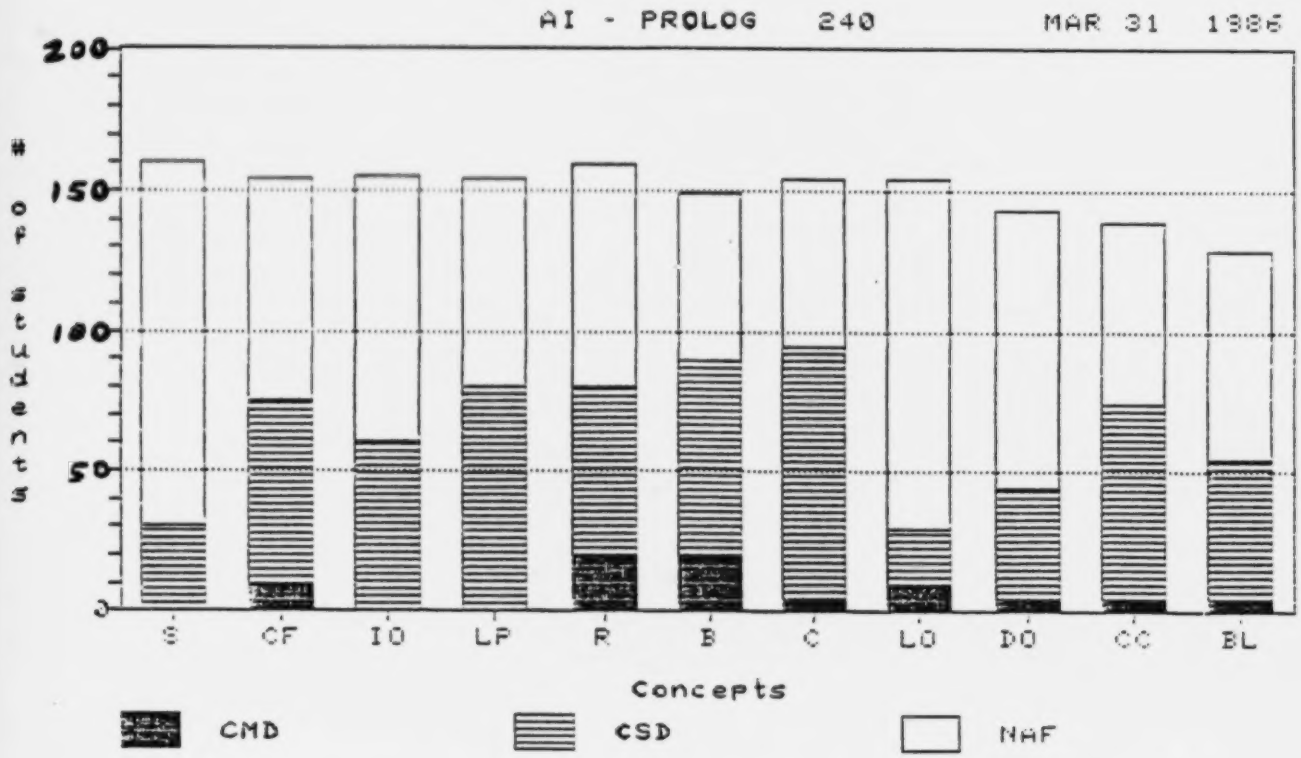

AI - PROLOG 240 MAR 31 1986
# of students
200
150
100
50
0
S
CF
IO
LP
R
B
C
LO
DO
CC
BL
Concepts
CMD
CSD
NAF

FIGURE 5(1)

CMD means caused the most difficulty
CSD means caused some difficulty
NAF means not a factor

S	=	syntax	B	=	backtracking
CF	=	control flow	C	=	cut
IO	=	input/output	LO	=	logic operators
LP	=	list processing	DO	=	database operations
R	=	recursion	CC	=	complex conditions

BL is an attempt to get the students to express their feelings about whether or not their background in other languages caused problems when learning prolog.

bottom space = continued influence from other languages.
middle space = problems subsided after a few weeks of using prolog.
top space = problems were noticed only with initial attempt to program with prolog.

es:

:) All students had experience in at least three procedural languages and none had experience in other declarative languages.

:) This was our first questionnaire.

FIGURE 5 (2)

References

Anderson, J.R., and B.J. Reiser, "The Lisp Tutor", Byte Magazine, April 1985.

Brown, J.S., and R. Burton, "Diagnostic Models for Procedural Bugs in Basic Mathematical Skills", Cognitive Sciences, Vol.2, pp. 155-192, 1978.

Burton R., and J.S. Brown, "An Investigation of Computer Coaching for Informal Learning Activities", Intelligent Tutoring Systems, in Sleeman and Brown (eds), Mass., 1982.

Carbonell, Jaime, "The Role of User Modeling in Natural Language Interface Design", in Applications in Artificial Intelligence, S.Andriole (ed), Petrocelli Books, Inc., Princeton, N.J., 1985.

City Software, "The NMR Simulator Operation Manual", City Software Development Corporation, Milwaukee, 1985.

Clancey, William, "Dialogue Management for a Rule-based Expert System: A Framework for Explanation", Stanford University, Report No. STAN-CS-81-896, Stanford, CA, 1981.

Cohen, R., and M. Jones, "Incorporating User Models into Expert Systems for Educational Diagnosis", Department of Computer Science, University of Waterloo, Research Report CS-86-37, September 1986.

Curtis, B., "By The Way, Did Anyone Study Any Real Programmers", Workshop on the Empirical Study of Programmers, 1986.

Hart, Anna, "Knowledge Acquisition for Expert Systems", McGraw-Hill, 1986.

Hartley, R.T., Coombs, M.J., and J.G. Stell, "Debugging User Conceptions of Interpretation Processes", Proceedings of the AAAI, pp. 303-307,1986.

Johnson, W.L., and E. Soloway, "PROUST: Knowledge-Based Program Understanding", IEEE Transactions on Software Engineering, Vol.SE-11, No.3, March 1985.

Kearsley, Greg, "Artificial Intelligence and Instruction: Applications and Methods, Addison-Wesley, California, 1987.

McCalla, G.I., Bunt, R. and J. Harms, "The Design of the SCENT Automated Advisor", Department of Computational Science, University of Saskatchewan, December 1985.

References (cont'd)

Michalski, R.S., J.G. Carbonell, and T. Mitchell (eds.), "Machine Learning" (esp. "An Overview of Machine Learning", pp.3-24), Tioga Publishing Company, 1983.

Sleeman, D., and J.S. Brown, "Intelligent Tutoring Systems", New York, Acedemic Press, 1982.

Van Arragon, P., "Using Scientific Theory Formation for User Modelling", Department of Computer Science, University of Waterloo, Ph.D Thesis Proposal, November, 1986.

Wahlster, W., and A. Kobsa, "Dialogue-Based User Models", in Proceedings of the IEEE: Special Issue on Natural Language Processing, Vol.74, No.7, July 1986.

Dr. Chris Nash
Director, Office for Educational Practice
University of Guelph

Which Objectives, Which Technologies? A Decision-Making Process

Two questions arise when any faculty member considers the use of an emergent technology for instruction. The first is a decision question, "How should I teach this so that my students will learn it?" or "What methods and emergent technologies should I use to promote student learning?" The second is an empirical question, "How is this best taught?" When the group supporting the use of an emergent technology is part of an office set-up to encourage scholarly approaches to teaching, it is important to make these implicit questions explicit as a first step in problem-solving. This explication is also the first step in a "curriculum analysis" process. The use of this process can lead a faculty member from trying an emergent technology in a single course to changing ways of thinking about the course and teaching in general. This paper describes the application of a decision-making framework to the question of "Which instructional technology should I use?"

The question of "How can I best teach this?" begins an evaluation of the relative effectiveness (in promoting learning) of two or more methods. This is the starting point in instructional design leading to a three-category typology of instructional objectives each with appropriate general teaching strategies (Robinson, Ross, white, 1985). This typology derives from literature on metacognitive strategies in teaching and learning, notably the work of Glaser (1978) and Brown et al. (1983).

The learning objectives for teaching are of two main types, cognitive and affective. Cognitive objectives have to do with developing student's knowledge of the subject. Affective objectives have to do with nurturing positive values, beliefs or attitudes about the subject matter or about the student as learner or practitioner of the subject. We generally espouse the goals of developing two kinds of knowledge learning, usually labelled "declarative" or "theoretical" knowledge (facts, concepts, theories) and "procedural" or "applied" knowledge (steps in intellectual or psychomotive procedures (Howard, 1987). Poorly designed courses which fail to take these distinct types of knowledge into account often fail to achieve their objectives, especially those of developing procedural knowledge. This is because procedures, whether of complex problem-solving such as experimental design, diagnosis, decision-making, or consultation or psychomotive clinical procedures are rarely learned by passive listening (Ross, 1980).

Effective procedural knowledge learning results from a series of instructional steps. The effective teacher of procedural knowledge begins with an explicit modelling of the procedure accompanied by a verbal description of the modelled procedures. This is followed by guided practice with constructive, timely and interactive feedback on

performance, followed by independent practice also with utilisable feedback on performance (Ross, 1980). A more elaborate model also includes the attention to the establishment of a theoretical knowledge base for the discussion including definitions of terms used to describe the procedure and its sub-routines such as "decision-making", "correlational stands" or "consultation", "hypothesis development" or "problem definition".

In contrast students' achievement of new concepts or the integration of concepts into more elegant theoretical explanations requires the use of instructional strategies of a different nature. Here we begin by promoting awareness of existing conceptions, and challenging those conceptions to produce an expanded knowledge, and finally integrating this conception into the student's larger theoretical knowledge (White, 1985).

The purpose of this paper is to use these basic ideas from curriculum analysis to describe a decision process about which emergent technologies to use. Then the "How best should I teach this?" question becomes the starting point for an evaluation of the effectiveness of the procedures chosen. Parenthetically, the paper illustrates in skeletal form the procedures for curriculum analysis described by Robinson et al. (1985).

The decision-making procedures are illustrated diagrammatically in matrix form. When an instructor is asking about an emergent, the first question for the design assistant is "What are the kinds of objectives to learning?" Is the instructor merely seeking to introduce variety into the method of presentation of factual material, or to make such material accessible to individual students out of class time? Or are there objectives for more complex learning such as concept development or growth in the ability to solve problems; design experiments or diagnose ailments. The latter abilities are examples of "complex intellectual skills" or "inquiry skills" (Robinson et al, 1985) and as such require skills learning. Concepts, on the other hand, develop best through interaction between students and instructor (White, 1985).

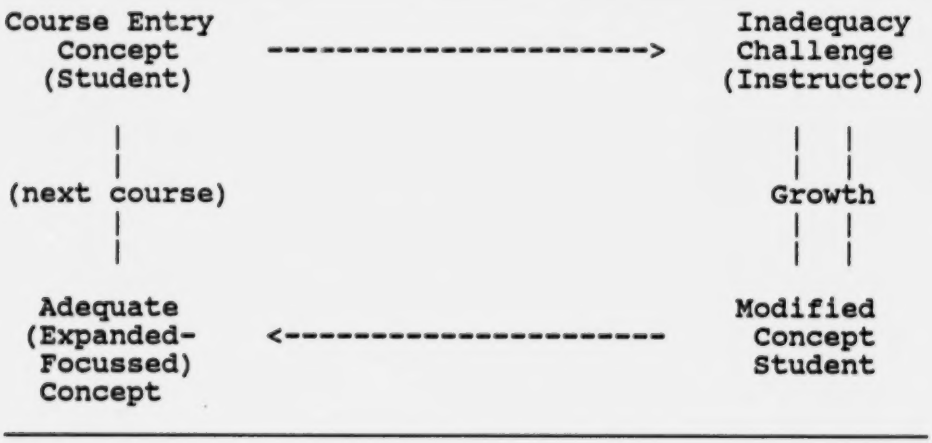
Figure 1, summarizes basic effective procedures for teaching students how to undertake complex (or simple) inquiries.

Figure 1: Inquiry Skills Teaching

. Modelling with description	Instructor
. Student practice with description	Student instructor
. Student practice with description	Student
. Student competence	Student

White encapsulates the results of his action research on those teaching procedures as "Instructor say, instructor do. Instructor say, student do. Student say, student do. Instructor get lost." (1983). Most instructors in higher education when they reflect on concept development where it is done well would elicit at least the basic steps outlined in Figure 2.

Figure 2: Interactive Concept Growth



White has elaborated this to the point where any professor working in philosophy of language would recognise his or her standard teaching procedures (1985), but the simpler diagram will serve our purposes.

Taking the inquiry skills, Figure 3 suggests a method the instructor might use to consider the characteristics of teaching required at each phase of teaching, from introduction of a procedure to its evaluation.

Figure 3: Identity Characteristics of Available Modes/Technologies for Instruction

	Professor/ Large Class	Small Class/ Lab	Film/Video	Other
Interactions/ Questions	MOD	HIGH	?	
Feedback				
Hand-on				
Practice				
Repetition				
Consistency				
Cost:Time Money				
Equipment Availability				
Instructor's Confidence or Ability to use				

These requirements are then matched to available modes of instruction such as classroom, small group or laboratory, programmed learning. Figure 4 illustrates this.

Figure 4: Identify Criteria for Phases of Instruction (e.g., Inquiry Skills)

Criteria-Needed for:	Phase: Modelling/ Describing Procedure	Guided Performance	Practice	Evaluation
Interaction/ Questions	MOD	MOD	LOW	NONE
Feedback to Student	NONE	HIGH	MOD	MOD
Hands-on Practice Accessibility?	?			
Simulation Practice				
Repetition				
Consistency of Instruction	HIGH	HIGH	MOD	?
Etc.				

When the faculty member has considered these requirements and available modes of instruction, the decision to use a specific technology is easy. Furthermore, attention has been given to basic notions of effective instruction, thus avoiding the pitfall of using alternative technologies to replicate ineffective instruction. As well, the creative use of individual technologies and conjoint use of various methods are stimulated. Similar steps can be taken for instruction aimed at developing or integrating students' concepts. Such approaches are possible when technological support staff are knowledgeable about empirical research about effective instruction in various contexts.

As to the affective objectives, the use of the most appropriate methods serves to motivate students in two ways. The more obvious motivation of novelty may be the less important. As educators we are becoming more and more aware that our students are partly motivated to learn or at least not put off learning, when they see that they are becoming more skilled or knowledgeable in the subject of study. So attention to providing better instruction in cognitive learning, paradoxically can produce gains in the affective domain (e.g., Nash, 1976;

Augustine, 1978). Students gain confidence in themselves as learners and a liking (or at least reduction of disliking) for the discipline as they become aware of their own competence at learning.

In conclusion, this outline is not intended to be prescriptive but merely illustrative of one potentially useful procedure for answering the question "which objectives, which technologies?". Its purpose was simply to suggest that faculty and technology support staff use a dual perspective when considering the use of emergent technologies. One perspective derives from knowledge about the capability of the technology. The second comes from a consideration of educational research into effective instruction anchored in metacognitive theory. Potentially the integration of the two perspectives can enhance the quality of instruction in both the specific course and subsequent course development by the faculty member. So our initial question expands the term "technology" to add the technologies or procedures of educational design. Empirical data about how best to teach for different objectives are used as the bases for the decision "What methods and emergent technologies should I use to promote student learning?"

References

Augustine, H. Congruence Approaches to Implementation. Toronto, ONTERIS.

Brown, A., Bransford, J.D., Ferrara, R.A. and Campione, J.C. "Learning, remembering and understanding, in J.H. Flavell, E.M. Markham (Eds.), Carmichael's Manual of Child Psychology, (Vol.1, 77-166). New York:Wiley.

Glaser, R., Advances in Instructional Psychology. Willsdale, N.S.: Erlbaum. 1978.

Howard, D. Feedback Systems in Distance Education. Paper presented at 15th Annual CSSE Conference, Hamilton, Ontario. June, 1987.

Nash, C. Teaching of French Through Geographie. Toronto:Ministry of Education. 1976.

Robinson, F., Ross, J. and White, F. Curriculum Analysis for the Improvement of Instruction. Toronto:OISE Press. 1985.

Ross, J. and Mayner, F. Teaching Problem-Solving. Toronto:OISE Press. 1980.

White, F. Teaching of Concept Development. St. Catharines, Ontario, Lincoln County Board of Education. 1985.

White, F. Interactive Concept Teaching. Paper presented at Northeastern Curricula Association Conference. North Bay. 1983.

LIBRARY_INSTRUCTIONAL_MODULE

ABSTRACT:

The University of Guelph Library's computer-assisted instructional package employs the VITAL (Versatile Interactive Teaching and Learning) software package. The Library modules form a generic, inter-disciplinary package designed to meet the needs of a wide variety of academic library users. The six basic modules offered range from the novice level ("Library Hours", "Library News", "Guide to the Building", "Borrowers and Borrowing") to the level of more sophisticated library users (interactional modules on the "Online Catalogue" and "Library Research").

Keywords: LIBRARY INSTRUCTION - CAI,
LIBRARY ORIENTATION - CAI,
VITAL

PLANNING:

In January, 1986, under the direction of the library's "Orientation and Bibliographic Instruction Committee", a task force was formed within the University of Guelph Library, to create an instructional/orientation package using the VITAL (Versatile Interactive Teaching and Learning) system. VITAL software was developed by Tayson Information Technology in conjunction with the "Instructional Technology Support Group" under the Office for Educational Practice at the University of Guelph. A detailed description of VITAL is included in these Proceedings.

The VITAL task force was made up of four reader service librarians. Although some knowledge of microcomputers was desirable, task force members were chosen primarily because of their experience in providing reference service in an academic setting, and because of their ongoing participation in the development and provision of a variety of library instructional services. It was hoped that direct involvement of the "purchasers" of the system in its development would both reduce costs and result in a better final product. Unlike earlier classroom teaching modules developed at Guelph (where graphic artists created screens designed on paper by course instructors), the library project was intended to be carried out by the end-users themselves - planning, design, creation, and revision of the module was performed by library staff, in consultation with the Instructional Technology Support Group. Since they were closely involved both with the development of the system and with the eventual users of the system, library staff were in a unique

position for monitoring and "fine-tuning" the instructional package in a responsive and efficient manner. The success of the library VITAL "test" project has in fact set somewhat of a precedent at Guelph -the Instructional Technology Support Group now routinely trains faculty and teaching assistants in methods of frame creation. The Instructional Technology Support Group itself produces only complex graphics, and concentrates instead on training of and consultation with VITAL user groups in the instructional and visual design of their modules.

The decision to develop a VITAL module for the University of Guelph library was motivated by several concerns. It was intended that a computer-assisted instructional system would enhance and supplement the library's regularly-offered building tours and class instructional sessions. The system could be used to reinforce what students learned in library instruction sessions. Usage of the module could be monitored, and would in turn help the library to develop and refine other orientation programmes. As well, it was hoped that a self-instructional system would offer a minimal level of instruction to library users during "off" hours, when reader service staff are not available to provide personalized instruction. It was thought that such a service would be of especial use to off-campus and continuing education students, and to non-university visitors, who generally cannot come in to the library during regular reader-service hours.

TRAINING:

At two introductory sessions, members of the task force were given basic instruction by Instructional Technology Support Group staff in VITAL sign-on procedures, basic commands, methods of frame creation, and general design principles. After this introduction, the librarians began creating frames. Each librarian devoted approximately fifty to sixty hours to preparing the first phase of the library module.

Initially, the learning process was difficult and slow, as none of the librarians had significant experience in either graphics design or systems technology. During this period, learning centred on the mechanical aspects of the VITAL software. A basic mastery of colour graphics technique was acquired - the effective use of colours, overlays, and text presentation all had to be learned.

Approximately twenty to twenty-five of the total 120 frames which made up the final library modules were created by Instructional Technology Support Group staff. These were incorporated in the library package by the library staff, with minor modifications and some additional text. Some of these frames were simple re-workings of frames created several years

earlier as part of the library's participation in a Telidon test-project during INET field trials. Others of these "created" frames were felt to be essential to the module, but of too complex a design for library staff to attempt creating. Initial costs to the library, then, consisted of a fee for the training sessions, charges for frames created by Universitel staff, and the "hidden" costs of library staff time. Since most of the frame creation was carried out by library staff, however, direct costs were reduced greatly. The major investment on the part of the library was that of purchasing the equipment needed to run the VITAL system. The module runs on a stand-alone system comprised of a PC (with a twenty megabyte hard disk), a colour monitor (with a NORPAK PCD6 colour card), and a TOAD (Teaching on a Disk) disk.

Once trained, the librarians were able to continue frame creation independently, while maintaining informal contact with Instructional Technology Support Group staff, who now acted as resource persons: giving advice about the aesthetic and pedagogical qualities of the module, as well as on hardware and software matters.

IMPLEMENTATION:

In operation, VITAL has proved to be a flexible system, adaptable to a wide range of instructional applications. One of its most useful features is the self-test instructional module, in which the instructor creates graphics frames and assigns questions related to the illustrations or to the subject matter.

Students answer these questions on the computer, and are instantly told whether their answer is correct or not. At the end of the session, they receive a score. In a library setting, the "score" is not reported to the user - this feature is obviously of more utility (and more acceptable) in a classroom setting than in a public service context. Library users are encouraged, but not required, to apply newly-learned informational skills. Unlike the typical classroom module, the library version of VITAL makes less use of self-test frames, and more use of "reference"frames which display fixed, assigned information. These frames are assembled in a logical, tree-structured sequence; the user selects from the main index and follows on-screen instructions to view information, which is usually a combination of text and graphics.

The library modules form a generic, inter-disciplinary package designed to serve all types of library users. There are four information modules and two interactive modules, one of which includes a question/answer segment. The four informational or reference modules provide such basic facts as library hours, floor maps, subject browsing, and borrowing regulations. This

information complements other types of information sources in the library; for example, subject information sheets, wall maps, guided tours, and an audio-cassette tour of the library. There are cross-references in VITAL to these other sources; VITAL is in turn "advertised" during library tours.

On individual module frames, information is presented in poster format, with essential elements highlighted, and textual details reduced to a minimum. In the planning stages, the text and illustrations were worked out on standardized "Entry Record" sheets, which provided a grid for designing graphics, and spaces for recording routing and keyword specifications. A flow chart was then drawn to determine routing and sequencing of frames. Experience proved the importance of well-planned routing, since it allows users to browse through the information package with ease, and to find quickly and directly the information they are seeking. When possible, users are always given the option of returning to a previous page, to the next page, to the main menu, or of exiting from the system completely. As it stands now, the system is self-explanatory and runs without personalized assistance. Minimal operating instructions are posted beside the terminal.

After being designed on paper,frames were created using VITAL software, stored temporarily as DOS files, then uploaded into the public library module. Frames can be easily updated, changed, or deleted using VITAL software.

CONTENTS:

In May, 1986, the VITAL equipment was set up for public use on the main floor of the library, next to the main Information Desk. During the first phase, 1986-87, the following menu choices were available:

1 LIBRARY HOURS

2 LIBRARY NEWS

3 GUIDE TO THE BUILDING

4 ONLINE CATALOGUE

5 BORROWERS/BORROWING

6 LIBRARY RESEARCH

LIBRARY HOURS and LIBRARY NEWS are simple informational modules which are frequently changed and updated as library hours,

orientation tours, and special events alter. GUIDE TO THE BUILDING is the largest module. Using floor maps, this package outlines the physical layout of the library buildings, and describes a variety of library services. The ONLINE CATALOGUE segment summarizes the main features and capabilities of the library's online catalogue and circulation system, and includes a short quiz tailored to the use of these resources. BORROWERS/BORROWING offers a brief description of the library's borrowing and inter-library services. LIBRARY RESEARCH explains how to use abstracts, indexes, and the library's catalogues to find information in books and journals.

PUBLIC USE:

The opening screen of the library version of VITAL is a picture of the Library building, with an on-screen invitation to "Press Return" to begin. Whenever there is no interaction on the system for three minutes, VITAL automatically returns to this opening page. This "billboarding" attracts the attention of passersby, so that the system is self-advertising. The library Vital system is available to any library user, and runs during all hours the library is open. No password is required to enter the system. However, the Library Instructional Module has also
been made available at several locations throughout the campus via the University of Guelph's communications link to the VITAL courseware database. Access to the module at the remote terminals requires an individual I.D. number and password, and thus is largely limited to students who use VITAL as part of their coursework, or to demonstrations of VITAL by the Instructional Technology Support Group.

In order to obtain feedback from users, a form labelled "Comments" was placed next to the VITAL library terminal as soon as it was made public. These forms have been collected regularly, and responses from library staff have been posted when appropriate. Comments to date indicate that while some users have found the electronic presentation of information rather elementary and unsophisticated, the general impression has been favourable. Users' opinions seem to be generally based on first impressions, and, for most of them, upon their first encounter with an interactive information system. Initial feedback showed that users were having difficulty logging onto the system, and some difficulty with browsing through the various menu levels. Since the logon procedure has been streamlined, and the routing of menu choices (hidden and public) expanded and simplified, the number of complaints of this type has dropped to a minimum.

FUTURE PLANS:

Introducing a special application of the relatively new videotext technology to an academic library setting has proved to be a challenging task. Some of the initial challenges have been met - library staff are now relatively confident in their abilities to design, create, route, and modify the VITAL screens; improved signage and routing have reduced user frustrations; and modifications to the VITAL software have made creation and manipulation of graphics and text much easier than it was at the beginning of the library project. More importantly, the potential applications for VITAL in a library setting are promising, and largely untapped. The system will continue to be adapted to changes in library policy, procedures, and philosophy of bibliographic instruction. A self-test module is already being planned to aid instruction in the use of the library's new CD-ROM catalogue. Additional modules are also being developed for bibliographic instruction in specific academic subject areas, and for graduate research. Short, subject-relevant library segments may be added as part of VITAL modules being used in specific courses at Guelph. All of these additions to the library's VITAL package will be interactive in nature, so that user interest (and therefore learning) is maximized.

The biggest challenge in developing the Library Instructional Module in its next phase will be to exploit what VITAL does best - sophisticated graphics effects, and a potential for <u>interactive</u> learning. Both of these elements are highly desirable in library instruction, but very difficult to attain through the traditional library instructional modes of signage, paper hand-outs, and large-group tours or lectures. In summary, then, the University of Guelph Library's VITAL project has had a satisfying start, but has also left much room for adaptation and growth. Fortunately, library staff have learned much from their initial involvement in the development of the module, and the VITAL system itself is flexible enough to allow for the necessary growth and change.

June, 1987

Gabriel Pal,
Lorna Rourke,
Helen Salmon

- McLaughlin Library,
University of Guelph

Computer-Assisted Language Learning (CALL) at the University of Guelph

D. M. Paramskas
French Studies
Department of Languages and Literatures
College of Arts

INTRODUCTION

Language learning has had a long, and not always happy, association with educational technology, including television, language laboratories, film and other media. The source of the problems can be summarized as an imbalance between hardware and software, equipment being acquired before its implications for language learning were thoroughly explored, and before an adequate number of quality programs were available.

Most problems were eventually resolved: while live television did not work, videotape became an indispensable learning tool; language labs centering on automatic progression disappeared in favour of Media Centers for individualized learning.

CALL also went through a very discouraging phase when the only machines available were main frame or minicomputers. Programs written for these machines tended to emphasize rote learning and were soon discredited. The arrival of the micro computer, with its capacities for individual, easy access, colour/graphics capabilities and foreign character founts, changed the situation dramatically. The micro became one more efficient tool to support the language learning process and to provide a range of activities suited to different learning situations and differing learning problems.

Currently, CALL programs are of six main types: individual drill/tutorial practice and/or remedial practice for grammar and vocabulary; games which include text reconstruction and simulations; writing assistance programs; grammar learning programs; and a wide range of template programs (mini-authoring systems)which allow non-specialists to custom design discrete lessons in areas such as reading comprehension, dictation and vocabulary building reflecting very specific local needs (Paramskas 1986a).

In the research stages are natural language interactive programs using parsers and voice-recognition, interfaced with video discs which will in effect simulate real-life communicative activity in the second language.

The University of Guelph has been a participant in the development of several current programs for French as well as English, Spanish, German, Italian and Latin: CLEF, a complete drill/tutorial courseware package for basic French grammar; the templates PROMPT (for reading comprehension), DICTATE (for dictation) and LISTEN (for aural comprehension), each of which functions for several European languages. An original Guelph project, GRAMMA, a parser for natural language input for French, is in the research stage.

These projects and programs, all written for IBM and compatibles, are described in detail in the following sections.

CLEF (Computer-assisted Learning Exercises for French)

CLEF is a courseware package for remedial/tutorial practice at the beginning-intermediate level of French as a second language. It is designed to function as an interactive electronic workbook, replacing the traditional print media workbook, and consists of 62 lessons covering basic French grammar (see Appendix A). Each lesson is made up of 1) a grammar explanation; 2) vocabulary drill for words used in the exercises which might not be transparent to a typical Canadian anglophone; and 3) a set of 3-5 exercises. Colour and graphics are used for pedagogical purposes; exercise types range from fill-in-the-blanks, through simulations and text reconstruction.

One of the principal design objectives of CLEF was the production of a sophisticated, universal language learning program, since learning populations and their instructors in Canada often differ widely in their skills, approaches and training. A collaborative approach to design was the solution(Holmes, Mydlarski and Paramskas 1982): CLEF was written by a team of instructors and programmers at three universities and one school board: the universities of Guelph, Calgary and Western Ontario and the London (Ontario) Board of Education. Reviewers have commented on the result (Russell 1985): software which is as portable as possible in pedagogical terms, i.e., independent of specific course organisation and specific textbooks.

CLEF Grammar explanations are stated in "classical" or neutral terms; no particular teaching approach is favoured. Since each lesson drills one discrete grammar point, instructors are able to shuffle the order of the lessons to suit whatever text or method they are using. CLEF is textbook independent: the vocabulary relies on cognates and a selection of the most frequently used words. When other kinds of words occur, they are first presented in the introductory vocabulary drill. As a fail-safe, and to

deal with a user population other than the target one, there is a dictionary function within each exercise. Users may at any point type "?", plus the word they do not understand. The translation appears at the bottom of the screen. All comments to students are given in French, although instructions are in English for the first 20 lessons and in French thereafter, and the optional grammar explanations remain in English throughout.

The program is menu-driven and designed to be completely user-friendly especially for those who are completely unfamiliar with computers. Two-finger typing skill is adequate for use; one may move from any one section of the lesson to any other by using the "quit" function (one keystroke).

Computer responses to user errors are a key factor in any pedagogical software. The program must be able to anticipate common errors and give clues specifically designed to help users overcome that particular error. CLEF avoids as much as possible catchall responses ("Wrong, try again") where the input is significative.

Errors have been classified into three categories: flubs, non-significant errors and significant errors (Paramskas 1983). Flubs are the result of spastic fingers - extra spaces, trailing spaces, lack of spaces, and odd capitals. These are disregarded if the answer is correct, and are automatically rewritten. Non-significant errors include single missing letters, doubled letters and single extra letters. The response to these is "There is(are) extra letter(s)", "There is(are) letter(s) missing" or, simply, "A typing error?" Users are allowed two tries for each question, but non-significant errors are not counted.

Significant errors are those which touch upon the grammar point being drilled (i.e., verb endings, adjective agreements, etc.) In this case, a long list of anticipated common errors has been foreseen, and when such an error occurs, it triggers an appropriate message, i.e., "Attention, the subject is plural, not singular"; "The ending is right but the stem is wrong" etc. Non-significative input (zero or garbage input) receives the only catchall comment in the program: "Sorry, not correct, try again". If the correct answer is not produced on the second try, it is given to the user, and the next question presented. Any non-answered questions reoccur in a "revision" at the end of the exercise.

Messages for correct answers are varied, random and two-tiered, ranging from Gallic excess for a correct answer at first try ("Fantastique!" "Chapeau!") to a subdued "OK" or "C'est ça" on the second try.

Whenever possible, alternate answers are accepted as correct.

For example, in a lesson on time, 6:15 may be acceptably given as "six heures et quart; six heures quinze; dix-huit heures et quart; dix-huit heures quinze".

Learner motivation is an extremely important factor in successful language acquisition. Motivation generally decreases where the learner perceives him/herself as caught up in an inflexible system whose pace and feedback are determined by a "machine"(Gagné 1977). The pattern drills derived from the global-structuralist learning theories are good examples of demotivating, mechanical formulas. Motivation generally increases where the students see themselves in control of learning and have choices in pacing and access (Paramskas, Error correction Alberta). CLEF is designed to reinforce high motivation both by giving control of pace to the learner and through a series of options.

Student control of pacing takes two forms: access control and progression control. Access control simply means that exercises may be chosen from the menu listing in any order, may be repeated or skipped, and the user may exit exercises at any point and be returned to the main menu.

Progression control is a more complex issue. Early CAI materials often relied on an internal timer to guide learners: so many seconds for viewing this screen, so many seconds to arrive at the right answers, automatic branching if time was not respected. It was felt that speed of acquisition was a factor in learning language and deviation from a standard rate, regardless of the ability to produce right answers, indicated some sort of failure. Under the influence of newer theories called "communicative", practice materials became more learner centered (Paramskas and Thomas 1986). The CLEF system rarely uses a timer, but rather depends on a carriage return for progression. In effect, this means that the fast learner and the slow learner may each go at his/her own speed, without feeling penalized, rushed or bored.

In addition, a series of options were developed to reinforce user ability to organize pacing and access to materials individually. These include the translation and quit options mentioned above. Within each exercise, users may also recall a summary of the

grammar in point form without having to leave the exercise and access the more complete introductory grammar explanation.

CLEF has met all of its proposed learning objectives. However, there is one drawback: the system is rigid, as any printed textbook can also be called rigid. Unless instructors are programmers or have access to experts, they cannot change any item or insert materials of their own; and even if programmers were available, the amount of time needed to program and debug would be prohibitive. In addition, all instructors know that no system ever completely meets their needs unless they themselves have designed it. As a result, the universities of Guelph and Calgary went on to write software which allows instructors with no computing knowledge to develop their own materials: PROMPT DICTATE and LISTEN.

PROMPT (Program for Reading cOMPrehension - Teacher controlled)

In this Guelph/Calgary project, it was decided to focus on a basic skill which has been widely neglected in second language study: reading comprehension. Students have traditionally acquired this skill on their own, plodding through texts with the help of a dictionary. If their comprehension level was checked at all, it was done through post-tests, by which time it was too late to correct misinterpretations resulting from comprehension difficulties or misuse of dictionaries. In addition, current trends in second language teaching emphasize oral communication and pour most resources into this area, to the detriment of reading skills. Yet without solid reading skills, no student outside an immersion situation can hope to acquire vocabulary and structures beyond a survival level. Educational theorists and teachers see the need for individual, supervised practice so as to build up reading skills (Mydlarski 1987).

While communicative methodologies perceive comprehension as the very foundation of second language learning, time is simply not available for such a tutorial approach. One of the pedagogical "ideals" is to have access to individualized materials reflecting differing learner capabilities and interests, and/or particular approaches favoured by each instructor. In cruel reality, the time necessary to create suitable materials is such that most instructors are forced to rely on standardized, commercial materials aimed at a hypothetical "average" learner, and to set exercises in which feedback is also relatively limited, sometimes to just "right" or "wrong". With a template system, however, a teacher could select any written materials at a variety of registers, quickly type in his/her questions and comments, and stockpile them in the computer. The exercises would be easy to edit, thereby ensuring a longer life (Paramskas and Mydlarski 1984).

PROMPT consists of two dimensions: (1) a program of prompts directed to teachers who merely need to type in their material in natural language, and (2) an interpreter program which uses the material entered and displays it for student use.

PROMPT includes a number of generally accepted features to guide CALL authors: control of speed of progression, choice of target language (French, Italian, Spanish, English, German and Latin), choice of instructional strategy, capability to try out a lesson just created, additional instructions for naive authors, control of some of the options available to the student, capability to edit any part of the program, and the capability to skip ahead or to exit from the program.

The student side of the system includes student control options (such as control of speed of progression, ability to move forward or back or to exit, help features which access the dictionary or additional instructions) and a high degree of interaction with the help of graded hints in the form of a dialogue which directs learners towards the answer. This dialogue becomes an integral part of the learning process as students discover and practice the structure and lexicon of the target language through reading.

Two kinds of exercises are included in the PROMPT format. The first is a passage of text followed by comprehension questions of the multiple-choice type. A series of menu-driven programs enables the teacher/author to enter the passage of text along with questions, multiple-choice answers, and comments on each choice. The author may also enter definitions and/or translations of words into the program's dictionary. All routines for display, options and error mark-up are embedded in the new exercise automatically while it is being created. It is intended that the multiple-choice answers contain a "best" rather than simply a "right" answer in order to encourage the use of precise language. All of the choices may be more or less acceptable but if a student chooses an imprecise answer, the program initiates a dialogue to lead the learner to a more nuanced response.

The second format is a variation of the cloze exercise. The student must reconstitute a passage in which certain key words have been omitted. The teacher/author enters the passage using markers to flag the words that are to be omitted, and may also provide clues, alternate answers, roots (where appropriate) and special prompts for anticipated wrong answers. As students respond to the exercise, the program automatically checks for spelling errors as well as morphological errors such as wrong verb endings, and displays a suitable comment. As in the multiple-choice format, the author may also include entries in

the program's dictionary (up to 50 entries for each exercise, which has a maximum length of 4 screen pages).

An editing feature in PROMPT enables teachers to change any part of the exercise in seconds and to test the new lesson. In some cases, PROMPT displays menus which guide the author quickly to the exact place to be changed or updated. When the exercises are ready for student use, a lesson assembler program transfers selected exercises to a student diskette. Since the content is supplied by the teacher, PROMPT lessons are suitable for a wide range of students, from junior to post-secondary.

DICTATE

While current language learning approaches emphasize the oral aspects of the second language, software to support oral/aural learning is almost non-existent. Voice recognition and production software does exist, but its quality is such that no second language teacher would accept the models produced. Technology will eventually solve this problem also. In the meantime, and using relatively unsophisticated materials, some support exercises for aural skills can be designed.

Dictation is a venerable, if not positively ancient, technique for sharpening the ear to the sound patterns of the target language, and learning the coding necessary to express those sound patterns in writing. It is, however, an extremely labour consuming exercise on the part of the teacher who wishes to exploit all of the pedagogical possibilities of dictation: each student production must be checked carefully; ideally, specific comments should be attached to each error; and the results handed back to the learner while his/her memory of the exercise is fresh. It can be easily understood that most instructors either avoid the entire situation or resort to such less-than-ideal feedback procedures as student auto-correction or mutual correction. Even when the feedback is done by the instructor, comments on specific errors are rarely given.

In the template called DICTATE, another Guelph/Calgary project, simple technology is used to facilitate and perhaps even improve the traditional dictation exercise. The text is supplied by an audio device, such as an unattached cassette recorder, controlled manually by the student who, these days, has no difficulty in manipulating this type of machine. The student is told to listen to the first sentence on the tape and to type it in a box on the screen, called the work area. The student can edit the sentence and, when satisfied, press the enter key to have the sentence checked. The program highlights all errors which the student may then correct. Otherwise, help in the form of a clue (the next letter in the word being processed, for example) or the on-line glossary may be requested. Once the sentence is correct, it is

moved to the top of the screen and the work area is cleared to enable the student to continue the exercise with the next sentence (Mydlarski and Paramskas 1985).

In contrast to the procedure commonly used in pencil and paper dictation where correction comes after the dictation, error mark-up in DICTATE proceeds one sentence at a time. This offers several advantages. A lengthy dictation would require considerable typing and could become boring or discouraging for the student, particularly if the whole screen is lit up with errors (the electronic equivalent of the "bleeding" page). Moreover, sentence by sentence correction offers the possibility that a student might learn from a mistake in one sentence and avoid making the same error later on, if the word were to recur. It was also felt that, because a student cannot advance in the exercise until the current sentence is rendered correctly, a series of small successes, unlike the red ink effect, would maintain self-image and provide motivation to continue (Mydlarski 1987).

The authoring system for DICTATE operates much like PROMPT with regard to text and dictionary entry. Authors may choose an existing passage or create their own. As source material, teachers may select any audio tape or any written text to record on tape. Once the material is ready, it is entered into the template as on a typewriter; teachers/authors have only to provide a glossary and the predictable errors, with a comment attached to each error. If teachers/authors choose not to supply any anticipated errors with appropriate comments, the program will automatically proceed to check the spelling. DICTATE uses a combination of pattern mark-up and error anticipation. The mark-up pinpoints the location of the problem by highlighting incorrect characters. Dashes are inserted for each missing letter. The system adjusts for non-significant punctuation and capitalization. Unwanted spaces are ignored. If further analysis is desired, students request a clue. Clues are verbal explanations provided by either the system or the author. Other features of the authoring system are similar of those of PROMPT, described above.

LISTEN is a third Guelph/Calgary project whose aim is to foster content-based aural comprehension, by using a combination of the techniques of PROMPT and DICTATE. The text to be understood resides on an audio cassette tape or a video tape, with the monitor sitting next to the computer. Either oral or written questions are given to the student, who answers in writing using either a multiple choice format, or a fill-in-the-blanks format. As in the previous two templates, exercise design is very simple for the teacher, and feedback to the student is immediate and specific. The above mentioned options are also available: a dictionary, a glossary, clues. One of the clues would allow students to identify an unknown word by citing the nearest

previous one that he/she understands, and setting parameters to print the following words until the unknown one is reached, at which point the dictionary may be consulted if necessary.
GRAMMA (Grammanalyseur/Grammanalyser)

Most CALL software consists of a set environment which the student must manipulate; there is relatively little place for free input. Any natural language (free composition) production can presently be evaluated only by an instructor, who must provide feedback on several quite different levels: orthography, morphemic and syntactic errors, style, coherence, originality. In second language learning, this procedure usually results in a paper covered with ominous notations in dangerous looking red ink, in a student who may become paralyzed by the dimensions of the disaster, and in an instructor who feels very frustrated at having to correct basic errors, and to spend a great deal of valuable time doing it. If a program could be designed to use a computer as an expert system, a pre-filter, a sort of grammatical tutor which the student would consult before handing in the composition, the instructor could spend quality time on evaluating those elements of written production which deal with substance and ideas: style, coherence, originality.

For a computer to become a grammatical tutor, however, it must be taught to "understand" at least some parts of a natural language. A computer can do this by means of a programming technique called parsing, which states syntactical rules for analyzing input, along with variegated semantic networks which permit some sorting out of ambiguities.

A "parser" can thus be defined in its simplest terms as a computer program which can decipher natural language input to a greater or lesser degree. The different types of parsers are usually grouped according to function: artificial intelligence research (Feigenbaum and Barr 1982); machine translation (Melby 1984); automatic textual analysis for literary concordances (Lancashire 1986); games such as ZORK or ELIZA (Kossuth 1985, Underwood 1984); first or second language learning (Paramskas 1986b). The latter group is generally referred to as "pedagogical parsers" and is further subdivided into two types: programs which assist student writing (organisation of ideas, sentence length, subordinate clause excess, repetition of flagged words such as "thing"), and programs which judge student writing (feedback on misuse of morphology, grammar; some semantic judging).

The design, as well as the capabilities, of parsers vary widely. There is no parser program yet available that "understands" natural language both syntactically and semantically. No one, to this author's knowledge, has as yet managed to write a parser that can handle 100% of free input. Machine translation programs such as the one used for translation of weather data from French

to English and vice versa by the Canadian Federal government manage up to 80% accuracy in a very restricted semantic field; the other 20% must be taken care of by humans. An analytical parser such as DEREDEC (Plante 1986) can generate deep structures very accurately, provided that the user first does an initial superficial level tag of each word, and provided the sentence is a standard one. Generative parsers such as ILIAD (Bates 1981) which have a resident semantic network, can produce fairly complex sentences that are semantically correct, but the student can only manipulate the sentences generated, not initiate them. The prototype of pedagogical parsers, Writer's Workbench (Raye 1983) is in fact limited to applying readability indexes, flagging discrete words such as "thing", counting syntactical and semantic reoccurrences and judging according to a pre-set formula what constitutes "good" style. The only syntactical deviance it can spot is the split infinitive. For years, IBM has been working on a pedagogical parser whose initial objective was complete analysis of complex natural language input with critical comments on both style and content. It began as EPISTLE, then transformed itself into CRITIQUE (Jensen et al 1986). It has yet to appear on the market.

The main source of the difficulties specific to the design of pedagogical parsers is the attempt to deal with syntax and semantics simultaneously. One may hope that in the future a complete parser will be developed, but for the present a more modest approach might produce practical results.

Guelph's GRAMMA project seeks to develop a pedagogical parser limited to the analysis of the syntax produced by students of French as a second language. Its aim is to signal to the user basic syntactical deviances, allowing the writer to eliminate most of the typical errors produced during a second-language composition. Correction is not automatic: errors are flagged and it is up to the writer to correct. No semantic judgement is made: the writer could produce a Joycean, surrealist or nonsense text and GRAMMA would accept it as long as the structures are acceptable.

In dealing with a language such as French, which has great morphological redundancy in its written form, it was most practical to divide graphemes into two groups: closed and open. Closed categories are those made up of a relatively small number of discrete items. Determiners, for example, consist of a short list of items which can be stored in memory; the same can be said of prepositions and pronouns. Other categories are to a greater or lesser degree "open" - that is, capable of containing an excessively large number of discrete items - the noun, the descriptive adjective, much of the category of "verb". There is, for example, no feasible way of listing all the possibilities of "noun", so that the program looks for a "default" system which sets up probabilities (Paramskas 1987).

Having received a partial definition of graphemes in the form of lists, the parser program first scans a sentence and tags each item (defined as any series of letters preceded and followed by a space or punctuation) according to the closed categories known.

For non-identified items, a set of probability rules are applied: sequences in which the open categories co-occur with closed categories. For example, an adverb (the most common of which can be listed) can be defined as anything ending in -ment that is not preceded by a determiner. In a given sentence under analysis, there are a number of unknowns labeled X left over from a first scan. Probability rules in the form of if/then statements are applied: if previously tagged item Y occurs in front of unknown item X, X is presumed to be a noun, unless tagged item Z co-occurs. Approximately 20 pages of such statements can cover basic French grammar as a second language learner would tend to use it. One case of unresolved ambiguity for French occurs if a previously identified noun is followed by a descriptive adjective (open category) followed by a positive verb (open category) in the present indicative which is not an auxiliary or semi-auxiliary. The only solution, while not elegant, is to ask the user if the item after the noun is an adjective or a verb (definitions and examples are provided) and proceed with the analysis.

Once items are identified, relationship definition occurs. Relationships are of two types: "links": gender, number, person agreements, for example; and "scenarios": identification of "illegal" sequences and "illegal" links, which constitute the bulk of a learner's errors when writing in this second language.

In the first two passes of analysis, the parser tags each item in the sentence, without reference to links or scenarios. It will therefore accept "je voulait" or "de le garçon" or "Il faut que je va"; once all items are identified, the categories of relationships will be applied, and any resulting discrepancies reported to the user.

The initial version of GRAMMA has been designed and is currently being programmed. It is expected that the final version would develop as a result of a series of field tests, in which original student compositions would be parsed, and where the parsing failed, GRAMMA's rules would be amended.

CONCLUSION

The CALL projects which the University of Guelph has initiated or in which it has collaborated are characterized by a focus on learning needs, as opposed to research in programming techniques. There has been a progression from fairly rigid programs imitating to a certain extent print media models, through the development of tools which allow teachers to control CALL materials, to the concept of a completely interactive student/computer situation in which the learner defines and controls his/her own learning needs and in which the computer becomes an original language learning tool.

REFERENCES

Bates, Madeline. 1981. "Generative Tutorial Systems." (ILIAD) ERIC ED222196.

Feigenbaum, Edward, and Barr, Avron. 1982. The Handbook of Artificial Intelligence, Vol. I. Stanford: Heuristic Press.

Gagné, Robert. 1977. The Conditions of Learning, third ed. New York: Holt Rhinehart and Winston.

Holmes, Glyn and D. Mydlarski, D.M. Paramskas. 1982. "Co-operative Courseware Production: A Solution to the Problems of CAI?" in Conference Proceedings, ADCIS. Bellingham, Wash.: Western Washington University Computer Centre. Pp. 252-258.

Jensen, Karen and G.E. Heidorn, S.D. Richardson, N. Haas. 1986. "PNLP, PEG and Critique: Three Contributions to Computing in the Humanities" in Ian Lancashire, ed. Computers and the Humanities: Today's Research, Tomorrow's Teaching. Toronto: Centre for Computing in the Humanities, University of Toronto. Pp 183-192.

Kossuth, Karen. 1985. "Using the Adventure Formats for CALI." in CALICO Journal 3, 2, pp. 13-17

Lancashire, Ian. 1986. Computers and the Humanities: Today's Research, Tomorrow's Teaching. University of Toronto: Centre for Computing in the Humanities.

Melby, Alan. 1984. "Machine-Assisted Translation for SIG (Special Interest Group) Report." CALICO Journal 2, 2, pp. 6-7

Mydlarski, Donna. 1987 forthcoming. "Two Template Authoring Systems: PROMPT and DICTATE" in Flint Smith, ed., Modern Technology in Foreign Language Education: Practices and Projects. ACTFL Foreign Language Education Series.

Mydlarski, D. and D. M. Paramskas. 1985. "A Template System for Second Language Aural Comprehension" in CALICO Journal 3, 2, pp. 8-12.

Paramskas, D. M. 1983. "Courseware-Software Interfaces: Some Designs and Some Problems" in CALICO Journal (Computer-Assisted Language Learning and Instruction Consortium), 1, 3, pp. 4-6.

Paramskas, D. M. 1986. "Computer-Assisted Language Instruction and the Language Laboratory" in J.R. Larson, ed. Applications of Technology: Planning and Using Language Learning Centers CALICO Monograph Series No. 1. Provo, Utah: Brigham Young University Press. Pp. 50-62.

Paramskas, D. M. 1986. "Artificial Intelligence in Computer-Assisted Language Instruction" in The Canadian Modern Language Review, 42, 3, pp. 619-628.

Paramskas, D. M. 1987 forthcoming. "GRAMMA: a Parser Program for French as a Second Language" in Flint Smith, ed., Modern Technology in Foreign Language Education: Practices and Projects. ACTFL Foreign Language Education Series.

Paramskas, D. M. and D. Mydlarski. 1984. "PROMPT: A Template System for Second Language Reading Comprehension" in CALICO Journal 2, 5, pp. 3-8.

Paramskas, D. M. and Alain Thomas. 1986. "Le Labo de langues à l'heure des approches communicatives" with A. Thomas, in Mosaique II: Tendances et Pratiques actuelles en didactique du français language seconde. Série Les Cahiers de l'Association des professeurs de français des universités et collèges canadiens. Pp. 19-27.

Plante, Pierre. 1986. "DEREDEC" in Willard McCarty, ed., Software Fair Guide. Toronto: Centre for Computing in the Humanities, University of Toronto. Pp. 38-43.

Raye, Carol L. 1983. "Writer's Workbench System: Heralding a Revolution in Textual Analysis: in S. K. Burton and D. D. Short, eds. Sixth International Conference on Computers and the Humanities. Rockville, Md: Computer Science Press.Pp. 569-572.

Russell, John R. 1985. "Computer-assisted Learning Exercises for French." Northeast Conference Newsletter 10, pp. 39-40.

Underwood, John. 1984. Linguistics, Computers and the Language Teacher: A Communicative Approach. Rowley, MA: Newbury House Publishers.

APPENDIX A

CLEF - LESSON CONTENTS

1. The definite article
2. Subject pronouns
3. -er verbs
4. The imperative of -er verbs
5. The interrogative with est-ce que
6. The indefinite article
7. The negative (ne...pas)
8. être
9. -re verbs
10. The agreement of adjectives
11. The position of adjectives
12. -ir verbs
13. avoir
14. Expressions with avoir and être
15. Contractions with à and de
16. The possessive construction
17. Possessive adjectives
18. Numbers 1-100 and dates
19. aller and the futur proche
20. venir
21. Prepositions
22. Stress pronouns
23. faire and expressions with faire
24. The partitive
25. prendre, comprendre, apprendre
26. pouvoir, vouloir
27. The interrogative & demonstrative adjectives
28. Time
29. dire, écrire, lire
30. Interrogative inversion, interrog. adverbs
31. The past indefinite with avoir, and agreement
32. partir, sortir, dormir, sentir, servir
33. The past indefinite with être, and agreement
34. The direct object pronouns
35. savoir, connaître
36. The indirect object pronouns
37. mettre, remettre, permettre, promettre
38. Verbs with prepositions
39. y, en
40. devoir, recevoir
41. Qui est-ce qui/que, qu'est-ce qui/que
42. voir, boire, croire
43. Relative pronouns: qui, que, dont
44. Adverbs
45. Reflexive verbs, and agreement
46. Negatives: jamais, rien personne, etc.
47. conduire, rire, courir
48. The future
49. Verbs with irregular stems (envoyer, etc.)
50. The comparison of adjectives and adverbs
51. The imperfect tense
52. venir de; depuis + present/imperfect
53. souffrir, ouvrir, offrir, couvrir
54. The conditional
55. naître, mourir
56. The demonstrative pronouns
57. The present subjunctive: form
58. The present subjunctive: uses
59. Impersonal expressions (falloir, valoir etc.)
60. The pluperfect and the past conditional
61. faire causatif
62. The passive voice

Laurie Pearce
Department of English
University of Waterloo

In the fall of 1986, sixty-three students enrolled in English 251, a mandatory course in literary criticism, at the University of Waterloo. These students were enrolled in the Co-op program, a successful program which integrates practical work experience with academic studies. As usual, the students were informed of the requirements for the course - a minimum of three essays of 1500 to 2000 words on various literary works. However, these essays were to be written and submitted electronically, using the University's IBM mainframe computer, and would be marked and returned to students in the same way.

The decision to use a computer system in the course was based on a number of factors. The University certainly had facilities to accommodate the approach, as it provides extensive computer resources on a twenty-four hour basis to all students, as well as training and support for arts students in particular. Many resources, such as writing tools and on-line mail and messaging, were available only through the computer. Also, most of these students would soon discover that their work experience required the intelligent use of computers, and the requirement of having to use a computer for essential academic work was a valuable learning experience.

The students in the course came from a variety of computer backgrounds. Those who had taken a first year introductory course, ARTS 198, had exposure to all the tools they would be required to use. Many, however, had taken a first year Computer Science course which taught them how to use the Macintosh microcomputer, and others had personal or high school experience with microcomputers as well. This experience was not directly related to the world of the IBM mainframe.

To do their course assignments, students had to be able to log on to the IBM system and use the CMS

operating system, read, send, and respond to electronic mail, use a text editor, print files, and send and receive files. They were also encouraged to learn to use the electronic writing tools provided for their use.

The procedure students followed was to first log on to the system. Any mail which was sent to them, either by other students or by the instructor, marker, or teaching assistant, was immediately listed. They could choose to read the mail or postpone reading it to another time. The mail system offered a "reply" function which made it simple for students to respond immediately to mail messages. Students could keep their mail in a file, which was useful for mail concerning essay topics and class news which they needed to refer to later. They then ran the CMS editor (a number of versions were available) and either began entering an essay or editing an existing one. Once the essay was complete, they used a simple command to send a copy of it to the course account. The command responded with a message when the file had been successfully sent. Terminals and printers were available on-campus twenty-four hours a day, seven days a week.

Regular training sessions provided by the Arts Computing Office were promoted in class, and additional sessions were scheduled to meet the demand. A written document giving some instructions on CMS was distributed to supplement existing documents available to students from the Math and Computer department. Despite these preparations and ample warning of the course's computer requirement, the deadline of the first essay found many students unprepared.

The variety of problems students had with their first essay and, often, in subsequent essays, made some training issues become quite clear. One training concern is whether to concentrate on purely

practical matters or general computer concepts. Much of what beginning users are taught is of the "magic button" variety, where students are simply told what keys to press to get the desired reaction. Concentration is on how to do things, which is a common-sense approach, but often general information about the computer, the operating system, the programs, and the way these components communicate is neglected. As a result, students and are ill-prepared to understand what to do when things go wrong. Students who did not realize that they were running a program called the editor within an operating system called CMS on a terminal which communicated with a central computer in a specific way had no basis for explaining or understanding problems they encountered. Students became physically dependent on one particular type of terminal, or even one particular terminal, because they weren't able to recognize the relationship between the terminals and the computer system. Time was wasted and effort lost because students weren't aware that when they used the editor they were working on a copy of their file, not the original file, so inadvertent changes could be ignored and files recovered. It was clear that a combination of conceptual and practical information was essential.

Another problem was that the Arts Computing Office courses provided general information for a wide range of users. These particular students needed instruction only in what they needed to know how to do to meet the requirements of the course. As a general rule, there is always more than one way to do anything, but for the urgent and immediate need of these students, they required quick instruction in one reliable way to do their work. Sophisticated usage could come later, as the need for it arose.

Confusion arose from the "ask your neighbour" syndrome, where students would turn for aid to

someone else, who was working on a different system or editor or application, and whose instructions would only confuse them more. The potential for disaster was increased by the fact that many students wrote their essays in the early hours of the morning, when no support staff was available to rescue them. A more complete instruction and reference manual was prepared mid-term to try to solve these problems.

An aspect of training which would be valuable to include in future courses of this type is training in the use of the computer as a writing tool. There are many drawbacks and advantages to writing with a computer, and students would have been helped in both their proficiency with using the computer and with essay writing itself by some instruction in this area.

For example, students should have been actively encouraged to print their essays, even though they were not required to hand in a paper copy. Quite simply, it is impossible to proofread accurately on a terminal screen. Individual characters can be difficult to distinguish and the stylistic flow of a paper cannot be determined from viewing a few paragraphs at a time. They should have been warned of the dangers of freely using the fast and easy move, copy, and delete functions of the editor, which often lead to choppy writing. They should have been taught of the many uses of the "search and replace" function. This can be used not only to find errors, but to save on typing time. A student can simply specify a string (such as "WS" for "William Shakespeare") for a long or difficult repeated term, and then replace it throughout the document with one command.

The writing tools provided for student's use were designed to find spelling errors, the use of "to be" verbs, cliches, colloquial expressions, commonly mis-used words, and transition verbs. Few, if any,

students actually used these, and it was clear that simply documenting their existence was not enough to ensure their use.

For the marking of essays, the use of the computer sol' d a great many problems and made organization of work simple. The procedure the marker followed was to log on to the system, respond to any mail from students, and check the "reader" (a sort of electronic in-basket) for essays submitted. Essays were consistent in appearance, so all could be considered equally, without advantage to those students with access to devices to present their essays more attractively. No time had to be wasted puzzling over handwriting, by either party - essays and marking comments were both completely legible.

Essays were marked using the same editing program the students used to write them. Comments were inserted, in solid capital letters, beneath the line they referred to. Standard abbreviations (such as "SP" for spelling mistakes) were used, and a list of these was given to all students. Since the file was being edited, comments could be as long as necessary. The text below the comment them simply moved down to make room for the new material, so comments did not have to be squeezed into margins. Also, comments could be easily edited if the marker re-considered a criticism or question. Overall comments, the mark, and the initials of the marker were always placed on the bottom of the file. When most or all of the essays had been submitted, marked essays were returned to students. This involved re-naming essays so that the name reflected that they were marked and sending them back to the students. Students could then either print or display them to review the comments.

Administratively the use of the computer was very helpful. A utility called "CLASS MAIL" sent

information on deadlines, assignments, and general news to every student in the class. Students had only to log on to keep up to date with their course requirements, placing the responsibility clearly on the student. News and information was announced in class, but students who missed class were assured of keeping up to date on assignments and general news if they simply signed on to the computer. Students were each given an alias as well as their difficult computer user i.d. so that mail could be sent to, for example, "Kevin" rather than "KLKROEKER". It was easy to see how many essays had been submitted, since they were all kept in the course account, and the computer recorded who had submitted them and the time and date of submission. Notes about missing essays were send electronically to students, so that the marker had a record of who had been informed of a late paper. Marked essays were accompanied by a mail message to inform students that essays had been returned and to ask for confirmation that students received them. This was not strictly necessary, but a number of students managed to "lose" their essays in either sending or receiving them, so it was adopted as a safeguard.

The mail system also made it easy for students to communicate with the instructor or marker without a face-to-face confrontation or the difficulty of physically finding the individual.

There are many possibilities for future uses of the computer as a writing and marking tool. For example, instead of using abbreviations such as "SP" in marking comments, a macro could expand the abbreviations into complete comments once the marking is complete. An on-line tutorial about writing problems could be designed, and comments on essays could direct students to the section of the tutorial which deals with the particular problems they have demonstrated in their writing. A

spelling and grammar checker for the marker's use could be designed, which would highlight errors and count them so that the marker would get an idea of how careful the student had been, without having to mark each error. The Computer Aided Learning module on The Rape of the Lock, which provides an on-line version of the text and several layers of detailed information about word definitions could be expanded to include other texts. Students would be able to use the module to help prepare their essay, and copy portions of the text directly in to their essays.

This use of the computer for the writing and marking of students essays and the administration of a course was a successful experiment. It is being repeated with a new group of students in the same course, as it clearly provided many benefits for students and instructors alike.

Laurie Pearce

May 1, 1987

Jacques Raymond
Department of Computer Science
University of Ottawa

Abstract

This presentation reports on the availability of the Electronic Overhead Project (EOP) and its application in teleconferencing or tele education. This system allows a professor to create/edit (an electronic equivalent of) slides and present them to an audience. Since the whole system is microcomputer based, it is easy to provide remote modem control of the system thereby providing teleconferencing capabilities, including 2 way pointing and graphics.

The slide editor can be used to create and edit "computerized slides" which are an ASCII coded representation of the content of a transparency. This content can be text, graphics, or a combination of both and a scenario. Graphics may include bit mapped captured files, video scanners output, video digitizers or standard graphic drawings. The system presented here is a software program providing these functions.

Once a set of "slides" has been created, the program can be switched to the "presentation mode" for presenting the slide show to an audience. A video projector is normally used to replace the computer monitor in order to accommodate large classes or audiences. If a 2 way communication system via modem is installed, teleconferencing becomes available. Furthermore additional sites can be serviced via the use of a phone bridge. The systems provides low cost communication system for teaching to remote sites. Since the whole system is PC based, it uses only off the shelf hardware and software components. Cost savings can be made by using voice/data modems which allow to halve the requirements for long distance phone links. The system has been used successfully in several courses and many conference presentations. Testing is underway for using it with VSAT satellite technology for teaching in remote north Ontario universities. Another project under development will use this technology to link several universities in Europe and Africa to Ottawa University.

The editor status line at the bottom of the screen illustrates the 3 modes of editor operations (fig. 1). The user can switch between these modes at any time for easy text/graphic entry. Another function key suspends editing and allows the user to "show" the presentation, the same key switches back to edit mode for easy modifications or new additions to the slide show.

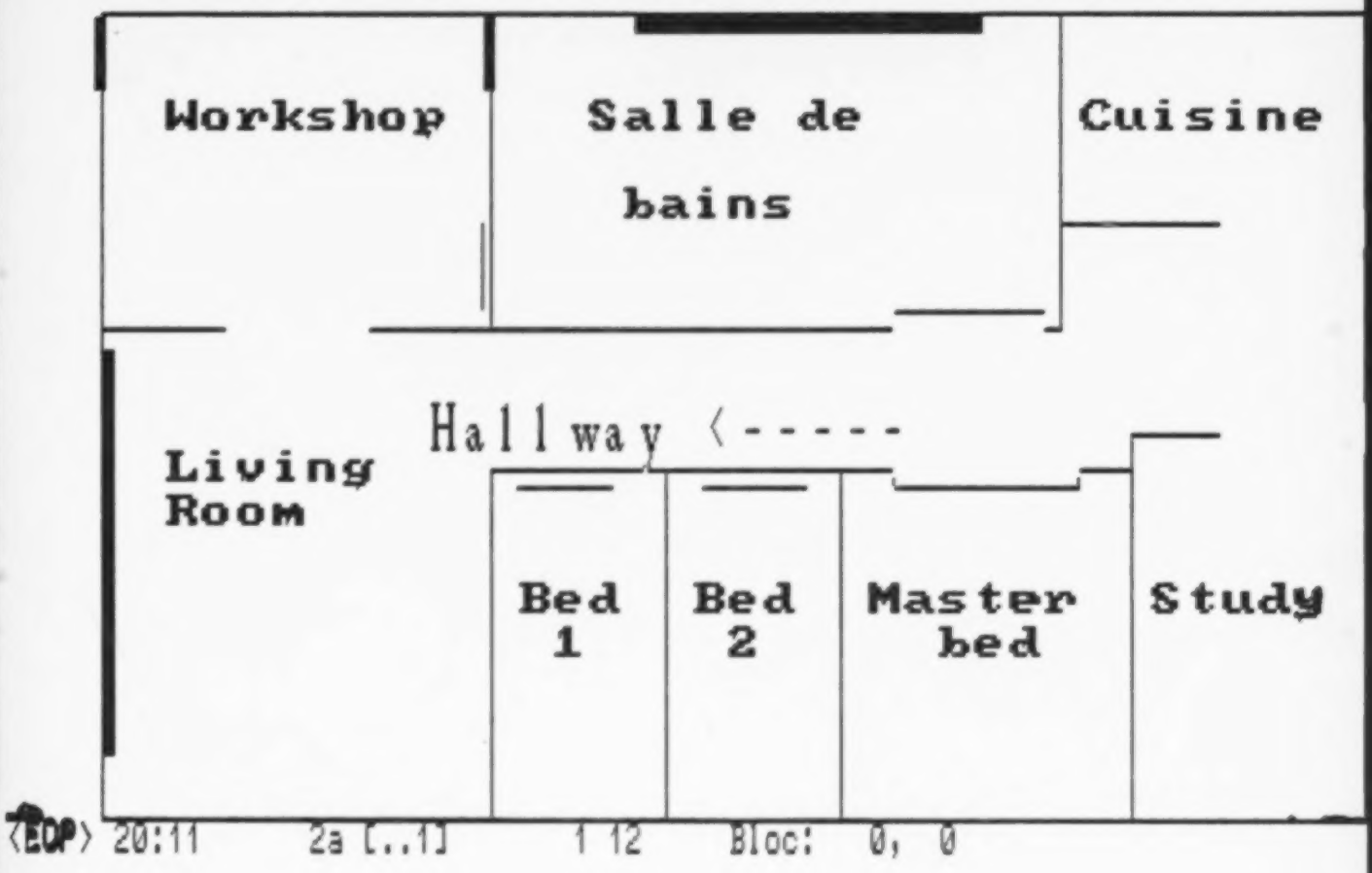

Workshop
Salle de bains
Cuisine
Hallway
Living Room
Bed 1
Bed 2
Master bed
Study
<EOP> 20:11 2a [..1] 1 12 Bloc: 0, 0

Figure 1

The 3 editor modes are now described. The editor provides the following editing possibilities.

a. Text editor

In this mode, the user has at his/her disposal a traditional text editor, its operations are very similar to Wordstar (TM) or similar editors. Although most standard text editor functions are supported there are some particularities to be found in this text editor since it was designed to edit transparencies as opposed to documents. These particular are now listed.

. Font specification: It is possible to specify (with a function key) one part of many character fonts, and see the text as it will be shown in the presentations. An unlimited number of font is available since the fonts can be created and imported from many commercial font editors.

. Up to 12 lines each slide has a maximum of 12 lines (or 6 of only large fonts are used). The line size is also fixed (40 or 20 according to the font used). Since the output is a set of slides, there is no need for document type specifications.

. Insert slide/delete slide commands functionnaly replace the traditional Page oriented commands. Carry over from one page to the next is not available, as it does not make sense while editing slides.

. Bloc commands (copy block, more block etc.) apply to a set of slides defined by its first and last slide, instead as of to a set of characters.

. A duplicate slide command is available for helping the creation of animated slides and animated sequences.

. Timing functions It is possible to associate an estimated time (in minutes) that one wishes to spend on a particular slide. These times can be modified, incremented, decremented at edit time. They allow the presenter software to (gently) guide the presentation as to speed limits and time constraints.

. Merging slide shows is possible in a manner very similar to the concatenation of documents with regular text editors.

. Compatibility with other editors. The same set of keystrokes as in Wordstar^{tm}, Turbo Pascal^{tm} or Turbo Prolog^{tm} editors is available, the user has therefore no need to learn new commands, in fact a complete slide show can be written on a regular text editor such as Wordstar^{tm} and the resulting document be imported into the EOP editor for presentation. Conversely a "slide show" document can be printed/edited using a text editor such as Wordstar. This presents no problem with text slide but can be cryptic with certain types of graphics.

b. The graphic editor

In this mode, and with the help of a mouse, a graphic table t, or a

keyboard, the user can design and enter simple graphics (rectangular frames or boxes, lines, free hand movements, circles etc.). The operation of the editor in this mode is similar to the one offered in software products such as MacPaint^{tm}, PC brush^{tm}, Dr. Halo^{tm} etc. Pictures can be superimposed to text. Any available frame grabber can be used to capture graphic output from most commercial packages and incorporate these captured graphics into the show. The capturing process condenses the file in the standard GWrite format providing substantial memory and floppy disk space savings. The sequence of image components can also be saved in order to recreate the image element by element during a presentation. If desired, other paint programs can be used to create images and their output can be imported into a slide show where the picture can be overlayed by text, comments, other pictures etc.

Another method for creating graphic is to define "graphic fonts" (for example icons) with a font editor, and then simply use the text mode to enter the desired drawing. This method requires a sizeable time investment for the design and the creation of the font, but proves to be much more helpful, practical and cheap in memory requirement especially if the whole content of a course will use the new font. For example a course in chemistry will benefit from using specific chemistry oriented icon fonts for its illustrations.

A third method for creating graphics can be found in the presenter software (not the editor) however we will describe it now. At edit time, a user can specify the name of a MS DOS program to be executed from within the slide show at a particular point in the presentation. This program can do anything the user needs at that time and in particular draw graphics, animations, simulations etc. etc. (fig. 2).

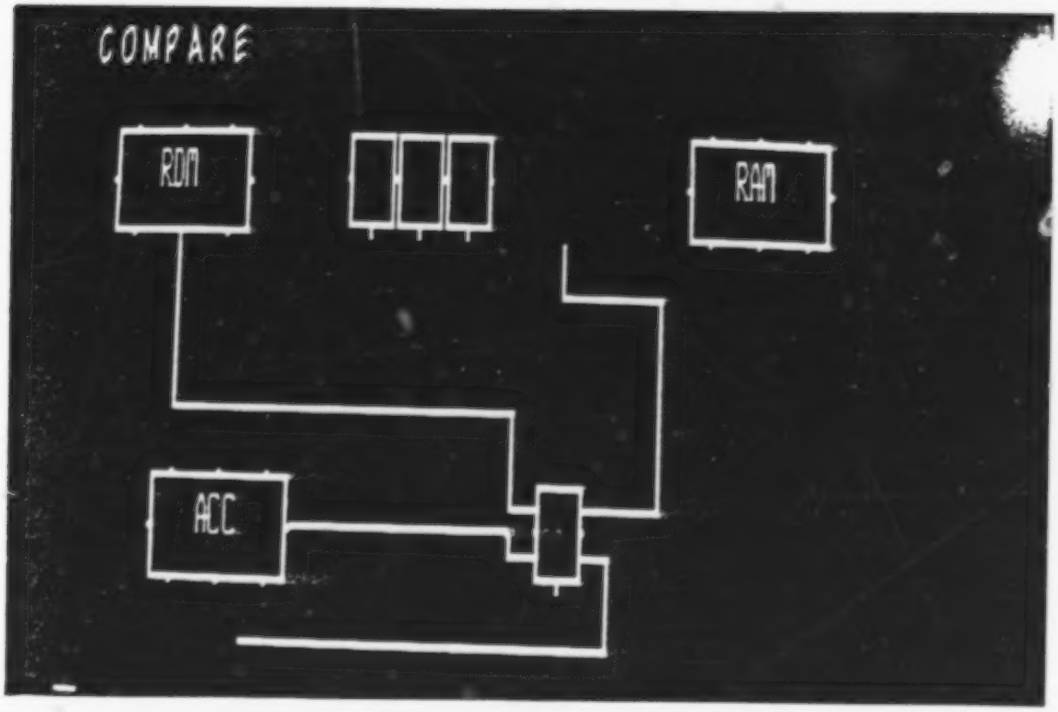


Figure 2

c. The scenario editor

Once the slides content has been decided and entered, the user usually would enter the edit scenario mode. In this mode he/she can specify the various animations and presentation functions available to each part of the slide or slides.

For example a slide (or part of it) can be shown, or hidden, or partially covered, or partially overlayed by another slide, or a line may be animated or blinking. Highlighting functions (frame, underline, blink, italics) can be invoked and its sequence in the show decided. Automatic time control can be specified (or manual control). Some of these functions will be described in following section on the presenter software since they are presentation functions, several dozens of these exist, and the user has probably more functions available than will be required.

Editing a scenario consists in editing 6 parameters for each part of a slide that is to be singled out as a "point".

1. Presentation function: What happens to the "covered point" when the "slide is put on the projector" (i.e. hidden, shown, blinking). A noop function means the point is ignored, while a view function means it is simply shown as is.

2. Explanation function: When the content of a slide has been shown (as per presentation functions) the slide scenario is scanned again point by point and a second command is processed. For example what was hidden can be uncovered, what was shown can be removed, overlaid, highlighted, etc.

3. Font. Which font is used to display this point.

4. Textcolor and background color for color cards and monitors if available.

6. function subparameter. This is used differently by the various functions, for example it may specify a font change, a duration in mn, the number of times an item is to blink etc.

d. Other functions

Once text, graphic and scenario are specified, standard file oriented functions permits the user to save on disk, concatenate shows, save and retrieve blocs of slides, look at directory etc. etc.

2. The presenter

The EOP was designed mainly as a tool to "present" a slide show. In this mode, a slide show is selected from the menu of presentations available and then either automatically or under the teacher's prompting the EOP will present each slide following the associated scenario.

If a presentation is reasonably formal (i.e. conference) the presenter simply presses the "Gotonext point" key on the keyboard or the remote control, and the EOP will follow the scenario as to what to present and how to present it. However at any time the user can choose any presentation

command he/she chooses to adapt to audience reaction, questions, or change of mood. Furthermore the slide sequence can be modified at anytime by using ESC and choosing the new (next) slide. This in effect overides the scenario. A "Oops" command permits to return to a previous slide if more explanation are required as requested by audience feedback.

At anytime during the presentation, the user can use an electronic pointer (a set of cursor controlled icons) with many different shapes, available in order to direct people's attention to particular portions of the slide. This pointer is controlled manually, can be removed, and can be copied onto the slide, in a manner analog to leaving pens, arrows etc. on a regular transparency.

Another manual function available during presentation is the scratch pad. When it is invoked, the screen is erased and a scratch pad is available for typing and displaying any text. If a mouse is used, free hand drawing is also available for sketching. The user returns to the interrupted slide with another keystroke.

During presentation, the built-in timing functions can alert the user of several problems: going too fast relatively to the planned schedule (or too slow), amount of time remaining and spent. Other functions include time-of-day and a stop watch for timing exercises or students.

Some of the most common presentation functions available are:

highlighting: frame, underlying, bold, move (animate) blink, italics

Transition: next slide, goto slide, oops, open window, strike out

animation: draw and hide, switch boards, do an animation, animate a line, switch line, hide line.

primitives: draw, hide, overlay, from top, from bottom, from left, from right.

system: change directory, load graphic, overlay graphic, point, change cursor shape, enter scratch pad mode, clean up overwritten comments, erase board, erase screens change board, stop watch.

Teleconferencing is no different than presenting to a local audience. Since the scenario (or changes to the scenario) processing is echoed to the serial port (with standard error correcting communication protocol), the slide show is exported to a remote PC running the same software.

Conclusion

The EOP was designed for classroom use and has been used often in this mode of operation. However with the single addition of modems the presenter can control more than one PC and thereby can show the course in many different locations. A very small bandwidth is required (300 bauds) since only scenario control has to be transmitted. This turns the system into a very cheap teleconferencing mechanism.

HORTICULTURAL PRODUCTION SYSTEMS PLANNING AND ANALYSIS

Jerald W. Riekels
Department of Horticultural Science
University of Guelph

In 1970, two new courses in horticulture were offered. These were named Production Systems (53-036) and Production Analysis (53-404). Any aspect of horticulture was subjected to basic exercises in planning and scheduling the work and materials required to grow horticultural crops or to render services of a horticultural nature. A financial analysis of each assignment was required also.

The original idea for these courses came from a horticulture course taken by the author in which each student planned an expensive and an inexpensive spray program for apple scab. This required the integration of knowledge of spray chemicals, disease characteristics, and growth characteristics of apple trees. Later in teaching a vegetable production course, this idea was expanded upon in a project in which the students prepared a weekly schedule of all the work, listed all the materials and equipment, and calculated the cost of growing a specific vegetable for one year.

Critical Path network methods of planning and scheduling activities and resources form the basis for the present courses. Arrow networks (Latterner, 1967) are convenient for establishing relationships between jobs, but line or bar (Mulvaney, 1969) networks represent the duration of each job to scale within a specific time frame (day or week). The resulting visual presentation (chart) is much easier for managers and workers to comprehend.

Network concepts were taught using slide transparencies. To avoid errors and waiting while students copy diagrams in class, a textbook (Riekels, 1971) containing prints of the diagrams was prepared. Later, in the first revision (Riekels, 1973), a general discussion of networks was added. The third revision (Riekels, 1983) was expanded to include detailed explanations of all diagrams and definitions. In the fourth edition (Riekels, 1984), some original ideas, specific to horticultural crops and services, were added to the basic network concepts.

In the early years of the courses, student assignments consisted of three separate production plans on three different topics plus a fourth assignment which integrated two or more topics to give maximum utilization and balance in scheduling workers and other resources. Three individual topics were reduced to two topics as similiarities were noted. Even later, only one commodity was required, but a second assignment was the incorporation of

alternative methods of producing the same commodity. The object of these single commodity assignments was to isolate the inputs, work, and costs pertaining to only one topic or commodity.

As the years passed, student inputs became more comprehensive gradually raising the standards and the content of the projects. Now, only one major assignment for two or more commodities is done, but the costs are subdivided for each commodity so the student can still learn how to assess individual crops within a more complex situation. The primary objective of the project is to create a plan that has a high probability of success if the work was attempted under real conditions.

Schedules of work, labor, resources, and crop growth for two or more horticultural commodities are shown in a bar chart representing one year of production. Job times (start, finish, and duration) are drawn to scale on the chart, and workers and other resources are listed with each job in a style that allows management personnel to see work alternatives easily and to balance jobs, workers, or other resources accurately and quickly. Employees can determine their work assignments easily also.

In the financial analysis, the total costs for each job are calculated and this forms the basis for cost control in the system. Individual job costs are simply added to determine the operating expenses per commodity or service. Very expensive jobs and job components can be identified easily for later adjustments to reduce the costs of an unprofitable crop or service.

Several years ago, an APL program (43 CPM) on critical path was used to teach network concepts. Unfortunately, this program could not be expanded to include all the work for one year in the production of horticultural commodities.

Recently, a computer spreadsheet (PC-CALC) has been incorporated to do the cost analysis of each production plan. A properly prepared spreadsheet with all formulas completely integrated will allow a manager to analyze any and all expenses at any time. Any anticipated change can be entered and the entire spreadsheet can be completely recalculated to determine the impact of one or more changes on the total cost of the system. Tables 1, 2, and 3 show portions of a spreadsheet for a project on golf course management.

The formulas for calculating the hourly cost of the of all fixed assets are stored in the appropriate cells in column F. The formulas are adjusted to calculate across each row pertaining to a specific piece of equipment, but the spreadsheet displays the results of the calculations in column F, not the formulas. A special spreadsheet command will display the formulas, if desired.

For example, the hourly cost of the tractor (Table 1, Row 22) is calculated by: ((B22-(0.05*B22))/C22/D22+E22+(.15*E22). This formula includes the purchase price (B22), the salvage value calculated at 5% (0.05*B22), the expected life in years (C22), the total number of hours of use per year (D22), the hourly fuel cost (E22), and an hourly maintainence estimate (15% of the fuel cost) (.15*E22). The annual depreciation is computed first, then, divided by the hours of use to get the basic hourly cost of the machine. After this is determined, the cost of fuel and an estimate for maintainence are added to get the complete hourly cost.

The hours of use (D22) are calculated by the formula (P17+P27+P37), which is entered into spreadsheet cell D22. This formula transfers the durations of those jobs which use the tractor, such as "SPRAYING FAIRWAYS" in Figure 2, from the specific job (P37); adds these durations together and puts the sum in cell D22.

To complete the cycle of calculations, the hourly cost to run the tractor is transferred to cells N17, N27, and N37 by entering the formula (F22) into those cells. In this manner, any change to the duration of any job using the tractor will result in the automatic recalculation of a new hourly cost of using the tractor (F22). This new hourly cost is automatically transferred to the appropriate cells in column N and new costs for the jobs are displayed in columns Q and R.

Costs of labor (cells T41 to T45, Table 3) and expendables (cells T6 to T13, Table 3) can be incorporated automatically into various activities in a similar fashion. Three activities in Table 2 utilize greenkeeper 2 at a wage of $9.00 per hour (T44, Table 3). If the formula (T44) is entered into cells N3, N8, and N36, then any change in the wages of greenkeeper 2 will be automatically transferred to these cells after the initial change is made in cell T44 of the spreadsheet.

In Table 2, the formulas in column Q multiply the quantities (column M) of each resource by the rate or price per unit (column N) and by the hours of use (column P). For expendable items, such as fertilizer (lines 24 and 29) and spray chemicals (lines 34 and 39), time is not a factor, so a value of "1" is entered in column P to complete the multiplication correctly. The cost of the tractor to "MOW ROUGHS" in row 17 is calculated by the formula: M17*N17*P17. Column R contains the costs of the jobs. Specifically, the formula in cell R18 is: SUM(Q16:Q18) which simply adds the costs of the resources for mowing roughs.

A manager can scan column R quickly and identify the most expensive jobs which should be analyzed first in order to reduce costs in the system. Then individual components (column Q) of an

activity can be examined and adjusted easily to create a less expensive production plan.

For example, a manager would likely reduce costs of "FERTILIZING" (Table 2, Rows 21 to 29) more by applying less fertilizer (cells M24 and M29) or by negotiating a lower price (cells N24 and N29) than by attempting to speed up the work to reduce the duration of the jobs. However, the labor to "MOW GREENS" (Table 2, cell Q3) is the most expensive component of that job. A manager could attempt to speed up that task by purchasing larger mowers to do the work in less time. As the duration of the job becomes less, the cost per hour of the mowers will increase. The manager can enter new mower prices into the spreadsheet (cells B5 and B6, Table 1), and repeatedly recalculate the spreadsheet until the least expensive balance of these resources is attained.

Once a spreadsheet has been completed to show all resources and activities, copies can be saved to show the work and resource utilization for each week, month, or quarter. Cost summaries, as shown in cells R41 to R44, can be calculated for any time interval in the system. Using this information, cash flow patterns can be determined easily and these patterns can form the basis for borrowing operating capital intelligently.

A spreadsheet which is completely cross-referenced permits a manager to enter a minimal amount of new information at any one time and then see the effect of the new information on production costs and profits almost immediately. Conceivably, a manager with a product to sell, can be negotiating the selling price with a purchaser over the phone while simultaneously entering the proposed prices into the spreadsheet. By the time spreadsheet finishes its calculations, the manager will know whether to accept or reject the proposed prices.

REFERENCES

LATTERNER, C. G., D. M. DRESDNER, J. A. SPIECH, AND G. M. USLAN. 1967. A Programmed Introduction to PERT (Program Evaluation and Review Technique. John Wiley & Sons, New York.

MULVANEY, J. E. 1969. Analysis Bar Charting - A Simplified Critical Path Analysis Technique. CRC Press. Cleveland.

RIEKELS, JERALD W. 1971. Horticultural Production Systems-Planning and Analysis. University of Guelph. First Edition.

RIEKELS, JERALD W. 1973. Horticultural Production Systems-Planning and Analysis. University of Guelph. Second Edition.

RIEKELS, JERALD W. 1983. Horticultural Production Systems-Planning and Analysis. University of Guelph. Third Edition.

RIEKELS, JERALD W. 1984. Horticultural Production Systems-Planning and Analysis. University of Guelph. Fourth Edition.

Table 1. Partial spreadsheet listing of equipment for a golf course including hourly costs.

	A	B	C	D	E	F
1	EQUIPMENT	PRICE	YEARS	HOURS USE	FUEL/HR	COST/HR
2						
3		MOWING				
4	GREENS					
5	Toro GM 3 A	$18,000.00	10	504	$.40	$3.85
6	Toro GM 3 B	$16,000.00	10	504	$.40	$3.48
7						
8	TEES & APRONS					
9	Jac.G.K.IV	$16,500.00	10	504	$.40	$3.57
10						
11	FAIRWAYS					
12	Ran.350D	$50,000.00	10	480	$.30	$10.24
13						
14	ROUGH					
15	Jac.(5G)	$12,000.00	10	720	$.00	$1.58
16	Rot.7'(PTO)	$2,000.00	10	96	$.00	$1.98
17						
18		TRUCKS & TRACTORS				
19						
20	Ford (55hp)	$15,000.00	10	176	$.40	$8.56
21	Massey Ferguson					
22	tractor	$12,000.00	10	832	$.45	$1.89
23	Chev 1/2 Ton	$11,000.00	10	4	$4.00	$265.85
24	Cushman B	$16,000.00	10	580	$.50	$3.20
25						
26		SPRAYERS & SPREADERS				
27						
28	Pro-Turf Sprayer					
29	(300gal)	$3,500.00	7	48	$.00	$9.90
30						
31	Hahn Fertilizer	$1,500.00	10	64	$.00	$2.23
32	Spreader					
33						
34		ACCESSORIES				
35						
36	Hole Cutters(1)	$75.00	2	100	$.00	$.36
37	Cup Setter(1)	$15.00	2	100	$.00	$.07
38	Hole Cups (20)	$5.00	2	1	$.00	$2.38
39						
40		HAND TOOLS				
41						
42	Shovels(3)	$15.00	5	115	$.00	$.02
43	Spades(6)	$20.00	5	104	$.00	$.04
44	Turf Edgers(2)	$15.00	5	42	$.00	$.07

Table 2. A partial spreadsheet listing of jobs and job components for a golf course.

	K	L	M	N	O	P	Q	R
1	JOB	RESOURSE	QTY	RATE	UNITS	HRS		JOB COST
2								
3	MOW	Grnkpr1	2	$9.00	/HR	504	$9,072.00	
4	GREENS	ToroGM3A	1	$3.85	/HR	504	$1,940.40	
5		ToroGM3B	1	$3.48	/HR	504	$1,753.92	$12,766.32
6	7days/wk x 3hrs/day x 24 wks							
7								
8	MOW TEES	Grnkpr1	1	$9.00	/HR	504	$4,536.00	
9	7days/wk	G.K IV	1	$3.57	/HR	504	$1,799.28	$6,335.28
10	4hrs/day x 24 wks							
11								
12	MOW	Grnkpr2	1	$7.00	/HR	480	$3,360.00	
13	FAIRWAYS	Ran350D	1	$10.24	/HR	480	$4,915.20	$8,275.20
14	5days/wk x 4hrs/day x 24 wks							
15								
16	MOW ROUGH	Grnkpr2	1	$7.00	/HR	720	$5,040.00	
17	5days/wk	MFerg	1	$1.89	/HR	720	$1,360.80	
18	6hrs/day	Jac 5Gng	1	$1.58	/HR	720	$1,137.60	$ 7,538.40
19	24 wks							
20								
21	FERT	Asst Sup	1	$14.00	/HR	32	$448.00	
22	GREENS &	CushmnB	1	$3.20	/HR	32	$102.40	
23	TEES	CSprdr	1	$2.23	/HR	32	$ 67.84	
24		20-5-8	100	$35.00	/25kg	1	$3,500.00	$4,118.24
25								
26	FERT	Asst Sup	1	$14.00	/HR	64	$896.00	
27	FAIRWAYS	MFerg	1	$1.89	/HR	64	$120.96	
28		HSprdr	10	$2.97	/HR	64	$142.72	
29		31-0-0	200	$25.00	/25kg	1	$5,000.00	$6,159.68
30								
31	SPRAY	Asst Sup	1	$14.00	/HR	24	$336.00	
32	GREENS	CushmnB	1	$3.20	/HR	24	$76.80	
33		CSprayr	1	$6.79	/HR	24	$162.96	
34		PCNB	20	$50.00	/25kg	1	$1,000.00	$1,575.76
35								
36	SPRAY	Grnkpr1	1	$9.00	/HR	48	$432.00	
37	FAIRWAYS	MFerg.	1	$1.89	/HR	48	$90.72	
38		TSprayr	1	$9.90	/HR	48	$475.20	
39		Killex	20	$15.00	/litre	1	$300.00	$1,297.92
40								
41					ACCESSORIES			$179.26
42					TOTAL OVERHEAD			$35,500.00
43					TOTAL OPERATING			$106,283.11
44					ANNUAL COSTS			$141,962.37

Table 3. A partial spreadsheet listing for expendable materials, labor, and general overhead including building costs for a golf course.

	S	T	U	V
1				
2		EXPENDABLE MATERIALS		
3				
4		PRICE	UNITS	
5	FERTILIZERS AND CHEMICALS			
6	22-0-16	$22.00	25kg	
7	31-0-0	$25.00	25kg	
8	20-5-8	$35.00	25kg	
9	Killex	$15.00	litre	
10	Terrachlor(PCNB)	$50.00	25kg	
11	Topdressing Sand	$40.00	cu. ft.	
12	Bunker Sand	$60.00	tonne	
13	Fuel	$ .45	litre	
14				
15		OVERHEAD ITEMS		
16				
17	TAXES			$8,000.00
18	INSURANCE			$3,000.00
19	ELECTRICITY			$4,000.00
20	WATER			$1,000.00
21	HEAT			$5,000.00
22	PHONE			$1,500.00
23	ADVERTISING			$3,000.00
24	FEES			$2,000.00
25				
26		PRICE	YEARS	
27	MAINTENANCE BUILDING	$180,000.00	20	$8,000.00
28	Woodframe, Metal Siding, Concrete floor, 4,200 sq ft.			
29	Includes:			
30	Office Space 10x20' (desk, filing cab.,chairs).			
31	Employee Area 20x30' (lunchroom, wshrm & showers).			
32	Service Area 20x30' (mech. tools, work benches, shelving).			
33	Equipment Storage 60x40' (Warehouse space).			
34	Fertilizer-Chemical-Seed Storage 20x20' (shelving, raised			
35	floor, ventilation).			
36				
37	TOTAL OVERHEAD			$35,500.00
38				
39		LABOR		
40				
41	TYPE	$/HOUR		
42	GREENS SUPERINTENDANT	$20.00		
43	ASSISTANT SUPERINTENDANT	$14.00		
44	GREENKEEPER 1	$9.00		
45	GREENKEEPER 2	$7.00		

THE IMPACT OF INTERACTIVE VIDEO TEACHING

Herminio Schmidt
Wilfrid Laurier University

The British Audio-Visual Association claims that we remember 10 % of what we read, 50 % of what we see and hear, but 90 % of what we say and do at the same time.(1) This revelation is of great importance to any educator. However, it has special significance to interactive video because interactive video covers all of the above learning combinations.

Interactive video is not an expensive toy conceived by business-oriented technocrats. It is a serious attempt by educators to deal with a new generation of students who have been raised and conditioned on a steady diet of color television. The average North American child sees about one thousand television commercials a week---roughly one million of them before s/he enters college or university(2). It stands to reason that today's generation of students has a much greater affinity to the TV medium than their parents ever had. For these reasons, more educators should take advantage of this powerful learning tool.

Business companies invest 20 billion dollars annually in advertising (3) because they have detailed proof that their message is effectively absorbed by the TV viewer (4). Language teachers have long realized the value of films and selected TV programmes as enrichment material. However, little research has been done in the field of interactive video learning as it pertains to language acquisition. Up to now we have neglected a very powerful learning tool.

Students who are entering our universities today have grown up along with technology. They have had their own TV sets; they have their own radio; some of them have their own computer. These students grew up with an interactive approach to their environment. If the program on TV is boring, or if they have heard enough music, they simply turn off the machine. Their interactions are based on an on-and-off response to a technological device. In a conventional classroom situation, today's students are tempted to say, "Can you fast-forward this lecture?". But this is exactly what the conventional instructor is not trained to do.

Sooner or later we have to adjust to this new participatory ethic. It is the educational manager who has to provide the means for this interactivity in the modern classroom. Today's technology has provided the means for us to adjust to these demands. A great opportunity for language instruction presents itself through interactive video because interactive video

responds best to the needs of this new generation of students. If we, as language teachers, do not utilize this medium more appropriately then students could turn us off too, like a boring TV show. Judging from the enthusiastic student response to the interactive VIDEO GERMANY program developed by Wilfrid Laurier University (5), we can say that they genuinely appreciate our effort.

The field-tested VIDEO GERMANY

The field-tested interactive video---we call it VIDEO GERMANY---was specifically developed for the language learner. It incorporates the highest level on the retention scale. Video Germany places a strong emphasis on the "say and do at the same time" (90 % retention) technique. It has clearly proven superior to other conventional it infotech-supported techniques, and has greatly increased student motivation.

In contrast to conventional video-supported learning, interactive Video Germany uses a different filming technique. The eye of the learner is the camera. It creates an illusion of an almost real situation. Instead of receptively watching the action on the screen, the learner in interactive video is the main character in the plot. He/she causes reactions and at the same time is constantly challenged to react by the persons on the monitor. This recording technique allows better utilization of the powerful TV medium for language instruction.

Interactive Video for the language learner should not be confused with talking to a machine like "Kit", the fully-computerized talking-car which long has been the favorite of North American children on TV's Knight Riders. Knight Rider will remain in the world of fiction for at least one more decade. However, as Paul Mermelstein, manager of Speech Recognition Systems for Bell Northern Research in Montreal recently said, "in terms of being able to give a set of restricted commands to some machines, we can do that today"(6).

This has been demonstrated by Mercedes Benz, the German car manufacturer, whose model 190E was shown at Expo 86 in Vancouver. The car will lock doors, open windows and adjust the angle of the rearview mirror. Separate verbal commands switch on the ignition, release or activate the handbrake. Even the radio responds to the driver's voice(7). Experts at the Symposium on Speech Recognition believe that within ten years we will be able to carry on limited simple conversations with a machine.

There is no need to wait ten years to incorporate interactive video into teaching. Interactive video has already proven itself in many fields through the videodisc systems. However, conspicuous by its absence among the already existing

interactive videodisc programs are interactive videos for the beginning language learner. This is understandable if we realize that even a totally integrated interactive (audio, video, computer) laserdisc system has its limits.

In order to get an interactive response from such a laserdisc system, the learner has to use a mechanical or electronic tool to generate a desired response on the TV monitor. This can either be a keyboard, a light pen, a mouse or a touchscreen. In any case the learner is required to make the decision by technical means in order to activate the program to its desired response.

This however, could be counter-productive for the learning of another language. Since speaking and writing are different learning skills, it is best to keep them apart in an interactive video approach. This separation of skills becomes significant especially when the aim is the creation of a realistic face-to-face environment.

We have therefore defined interactive video for the learner interested in a language approach that stresses the communication aspect. ''Interactive Video'', as it is referred to in this article means:

> a near-life communication between two or more persons via a TV screen whereby the learner is in front of the monitor. The verbal interaction should come as close as possible to a real person-to-person and face-to-face communication. No participant should be diverted by pressing keys or other gadgets which would reduce the illusion of reality.

The needs of a language learner are quite different from students learning any other subject, especially when the aim is to enhance the oral communicative skills. Pressing keys destroys the illusion of a real-life conversation. Therefore, in Video Germany we do not mix the skills nor the media. A switch from speaking to writing would amount to an emotional cold shower. Instead, the written exercises are kept entirely separate from the video episodes. With this technique the real-life illusion is kept intact. Furthermore, for Video Germany there is no need to press any buttons because the interactive stimuli have been incorporated directly into the software, the program.

In addition to the benefits to the students, there is another spin-off of equal importance. Because Video Germany can be played effectively in linear form as a video tape, it does not require all the extra electronic equipment necessary for a videodisc system. In other words, Video Germany is very economical. The main thrust of this interactive Video Germany approach however, remains the emotional and creative involvement of the learner. Sufficient evidence has been collected over the past two years which indicates that interactive Video Germany is a powerful

boost to student motivation and learning.

There is a distinct difference between a conventional video film and an interactive video. In a conventional video film the world is perceived and experienced through the eyes of the actors on the screen. The learner in front of the TV monitor remains a passive observer of a world which is presented by the actors on the screen.

In contrast, in interactive Video Germany the learner is the main character and therefore helps determine the action in the electronic world. The learner as the main character can therefore never be seen on the screen, because the eye of the learner is the camera which records the world. This recording technique allows better utilization of the TV medium for language instruction. It creates an illusion of an almost real situation. The main thrust of this interactive approach is the emotional and creative involvement of the learner.

In planning the Video Germany program we incorporated some basic principles from behavioral and cognitive psychology as well as the personality theory. Right from the beginning, we built in elements that challenge the student's natural need for survival. The student has to use his limited vocabulary to preserve his personal dignity. We consider this, in itself, to be a very important and meaningful exercise. It is called effort-stress. It not only promotes learning but also shapes the student's attitude towards coping in any difficult situation. Great importance is attached to the fact that with maximum fidelity the learning experience simulates the conditions under which the skill will be practiced in real life.

We all have the need to control our environment. Students become very anxious if they cannot control the language environment. In Video Germany we give students the opportunity to free themselves from anxiety by encouraging them to interrupt the speaker on the screen. In Video Germany students are urged to stop the speaker. They are encouraged to interrupt; they will ask the speaker to repeat or to speak slowly. Again, all this is incorporated into the program in order to give students a sense of control over their environment. This type of interactive technique creates self-confidence and leads to self-esteem which is a very powerful aspect for the successful language learner(8).

The interruption technique is especially important in a new language environment. The student has to be able to control the speaker(9). By being able to control the speaker the student will form a positive and healthy attitude toward the second language.

Video Germany consists of several parts. In the very first video lab the student is welcomed by a narrator while the title flashes

on the screen. Then the screen is automatically turned off while the narrator introduces the learner to the new approach. The audio-only introduction helps to ease away any emotional barrier the student might have. Since the Video Germany episodes are presented as the learner's trip to Germany, the student's willingness to believe in this illusion is prepared in this introduction. In a very calming and relaxing voice to the tune of especially selected baroque music the learner is first relaxed, s/he is then asked to close his/her eyes and is then eased into an imaginary trip in his/her own mind. This very enjoyable experience helps to break the ground for the acceptance of an exciting journey to Germany. After this introduction the video drills are presented.

The video drills are purposely arranged in a relatively sterile and conventional fashion. Here the student is being treated in the conventional manner, as a student who must endure some not-so-exciting learning tasks. Even though this video drill portion is intended to be a relatively sterile learning situation, students report pleasure in doing the drills. Most of the students consider the video drills beneficial. They feel it is helping them memorize and improve their pronunciation because they can clearly see the lip-movements of the speaker. Watching the movement of the lips is a natural and very effective technique for most language learner.

What does VIDEO GERMANY Contain?

In keeping with the underlying thought that the drill phases are to be considered as "work", we have distinctly separated them from the "reward": the Video Germany Episode. For the "real" trip to Germany, the learner is exposed to "real German" people and "real German" scenery. It is quite a realistic trip because everything is experienced from the perspective of the learner.

In order to maintain the "realistic" illusion, the episodes are never interrupted. It is felt that any interruption, at this stage, for comprehension check-up etc. is counter-productive because it destroys the illusion of reality. Students will adversely react to having their illusion destroyed. It would create an emotional barrier which affects the learning process.

During the real-life episode no correction is given other than what would be considered as part of a normal and natural reaction of a conversation partner. By strictly separating the "teaching" (drill, work) and the "real conversation" (reward) we have "matched" different approaches with the students' learning styles. The attention paid by Video Germany to the instructional situation exploits a variety of learning strategies which are most effective for most students. Individuals are able to learn "with their best abilities", as Diller has pointed out quite appropriately (10).

This underlying instructional approach has determined the different filming technique in Video Germany. It has determined the audio introduction as an ''easing-in'' of the student into a world of willing acceptance of the illusion, and every effort has been made to follow this illusion to its logical conclusion.

The episode begins, therefore, at the airport where the learner is greeted by a flight attendant and presented with a complimentary ticket to Germany. Then, in the waiting area the learner is approached by a helpless elderly lady who is returning to Germany from a visit to her son. Of course she does not speak English and therefore engages our video learner in a dialogue. The student is left to respond to all the questions. The narrator, who is never seen on the screen, acts as the learner's subconscious voice: he encourages him/her to respond and gives positive reinforcements. With the departure of the plane the first episode of Video Germany comes to an end.

The following episodes, ten in total, are basically the same in structure: first the different phases of the drills and then the Video Germany episode. In the process of the ten episodes the learner arrives in Frankfurt, asks for directions, takes a train to Marburg, finds a hotel, meets people and visits Marburg's modern university hospital. During the discussions with German friends a variety of subjects are dealt with, culminating in a Kaffee-and-Kuchen invitation by one of the friend's parents.

Quite often the learner is showered with information he cannot cope with. We provoke the learner to stop the speaker, to have the speaker repeat. With this technique we give the power to our student. We build self-confidence. Our student can do it! The feedback from the students is exclusively positive and enthusiastic. They feel that the real-life situations are challenging, interesting and most of all relevant. In the process, they experience and learn to cope with a variety of situations in Germany. They learn a great deal about Germany and they have to talk about their own country. All this is enhanced by the interactive approach which deeply involves the learner in the process.

Video Germany is independent of any textbook. However, it follows the basic development of the conventional college text. It starts with very basic elements and progresses step-by-step without making it obviously grammatical. It can, therefore, be easily integrated into a conventional beginners' course without dominating the existing curriculum. Each of the ten units of Video Germany lasts thirty minutes and includes the drill phases. In the language laboratory, the voices from the video and the learner are recorded on an audio tape. The tape can be replayed and corrected, if desired.

The comment expressed most often by students relates to the joy of learning with Video Germany. The enthusiasm of the students is based on the near-life situations, the personal and practical experience they gain through the interactive video approach. Students' motivation is considerably higher than those students who are only exposed to an audio language laboratory. In answer to the question, "What is your reaction to the Video Germany episode?" most students felt it was "excellent", and one student expressed the sentiment of many by saying,

"I really enjoyed it and I can't wait to arrive in Germany."

Sufficient evidence has been collected over the past two years that Video Germany is a powerful boost to student motivation. Moreover, Wilfrid Laurier University is thoroughly impressed by the results of the VIDEO GERMANY program. It is presently developing a second project, an interactive VIDEO FRANCE(11) program modeled after the Video Germany concept.

In conclusion, it can be said: the technology as well as the viability of the interactive video approach for the language learner is within our reach. We feel that we have made a significant step into a new era of effectively utilizing a very powerful tool for the benefit of a new generation of language learners. The interactive video-approach in fact has already demonstrated renewed interest in learning languages.

1) Eric Parsloe (Editor), Interactive Video, Cheshire 1985, p.161.

2) Gerhard Marletzke, ''Jugend '85, Full Bock auf Medien?'', ARD MAGAZIN, Nr.1, 1986.

3) US Bureau of Census. Statistical Abstract of the United States 1986, p. 552.

4) Sandra J. Ball-Rokeach, M.Rokeach, J.W. Grube, The Great American Value Test. Influencing Behavior and Belief Through Television, London 1984, p. 92.

5) Video Germany, an Interactive Language Course for the Video Language Laboratory. By Dr. Herminio Schmidt and William E. Nassau LRPS; Copyright by Wilfrid Laurier University, Waterloo, Ontario; Canada, N2L 3C5.

6) Computing Canada, Vol.12, No.16, p.9, August 7, 1986.

7) SCALA, 7-8/1986, p.6.

8) H.H. Stern, "What can we learn from the good language learner", Canadian Modern Language Review, Vol. 31, No.4, pp. 304-318.

9) Jerry Richardson, The Magic of Rapport, 1984, p.151.

10) Karl C. Diller. ''Criteria for adapting language teaching methods to the learning styles and abilities of students. Proceedings of the Fourth International Congress of Applied Linguistics, ed. Nickel, G. Stuttgart: Hochschulverlag, (1976), 3, pp.341-351.

11) Video France (S\'ejour en France) is based on the same concept as Video Germany. Video France is scheduled to be available in the fall of 1988. (For more information write to: Wilfrid Laurier University, Waterloo, Ontario; Canada, N2L 3C5).

THE USE OF THE VETERINARY MEDICAL INFORMATION MANAGEMENT SYSTEM (VMIMS) DATABASE IN TEACHING

T. Stirtzinger DVM PhD,
Department of Pathology,
Ontario Veterinary College,
University of Guelph,
Guelph, Ontario.

In recent years the rate of growth of medical knowledge has been greater than that at any other time in history and veterinary medicine has both contributed to and been affected by this phenomenon. At the level of undergraduate teaching the proliferation of knowledge has resulted in the need to include increasingly more factual information in the curriculum, consequently reducing the time which students have to integrate and synthesize that information and thus develop and practice those problem-solving skills essential to the provision of optimal patient care. There is a need in medical education and veterinary medical education to explore teaching strategies which foster both the acquisition of knowledge and its use in the clinical setting. Those techniques which encourage active participation by the student in the learning process are the most likely to be both effective and well received.

Recent advances in computer science and technology have paved the way for the introduction of computer assisted instruction (CAI) in medical and veterinary medical education. The increasing availability of large computerized data bases of medical literature has allowed for the frequent updating of factual information. Computer-based "drill and practice" programs can assist with the memorization of unambiguous factual material. Medical information systems allow data collected in the process of actual patient care to be used in the construction of case simulations which help students to develop problem solving skills. When coupled with techniques for self-assessment students can master the skills necessary for clinical competence in the manner and time-frame best suited to their individual abilities and preferences.

Classically, involvement in patient care has been the mechanism by which students have learned how to apply their factual knowledge to the solution of real-life problems. This approach, although it is the most direct, does have several limitations. Specialization is a fact of life at any large teaching institution; therefore the amount of time which a student spends in any one area is by necessity limited. Unless the case load for that area is unusually large or the student is exceptionally persistent, the chances are that any one student will not see the range of problems that he or she will be required to recognize and manage in practice. In today's society the veterinarian is

still looked upon as a generalist, able to deal with problems involving any body system in a wide range of species, and to keep in mind economic realities while doing it. Patient data stored in medical information systems (MIS) can be used to supplement, in a case-oriented manner, the experience which the student acquires during the process of patient care.

VMIMS, The Veterinary Medical Information Management System:

The Veterinary Medical Information Management System, VMIMS for short, was developed at the Ontario Veterinary College and was implemented in 1983 for the purposes of storing information concerning patients seen at the Veterinary Teaching Hospital. VMIMS operates under the DOS and MVS operating systems using the Sharp APL programming language and is able to run on an IBM (or compatible) computer of the 4300 model series or larger. Built into VMIMS is a security system which prevents illegal access or data entry by unauthorized individuals. In addition, safe-guards ensuring client confidentiality are an integral part of the system. In designing VMIMS its potential uses for hospital administration, research and teaching were all kept in mind. The system continues to evolve through the input of all those who currently work with it.

At present over 28,000 individual patient files are stored on the system. These patient files contain data derived from several areas. Clinical information such as patient characteristics, diagnosis, operations or treatment codes and outcome, as well as other information required for administrative purposes is entered for every patient. Further information is added as it is generated during the process of patient care from the areas of bacteriology, clinical pathology, radiology, and parasite.

On admission each patient receives a unique hospital number through which that patient's record can be accessed. The same hospital number is used for any subsequent visits. Information within each record is arranged according to visits, with the most recent visit available first. Information for the various areas can then be selectively examined through the use of a menu system and it is possible within any one area to "page" backward and forward through the patient record in order to follow changes which have taken place in a parameter of interest over time. In addition to the "show record " function used to examine individual patient records when the individual's hospital number is known, a "search and gather" function allows groups of cases to be selected according to user defined criteria. For example, it is possible to select for examination the complete records or any part thereof of all German Shepherd dogs aged 10 years or older, treated at the veterinary teaching hospital in 1986. If a specific disease syndrome is of interest all cases of that syndrome can be retrieved by specifying the appropriate diagnosis

code. Again, once cases of interest are selected any of the information available in the records can be printed in a chart form or the individual case records examined by using the hospital number and "show record" function. It is possible to perform a limited amount of statistical analysis within the system itself or alternatively to transfer the data to a microcomputer for more sophisticated analysis. A limitation of the system at this time is the lack of information from areas such as anatomic pathology and cytology, both of which contribute information which is often of major diagnostic importance. Neither is there any information recorded for patients seen on-farm, nor is herd health data stored under the VMIMS system. It is planned to bring all of these areas on-line in the near future.

The Use of VMIMS in Undergraduate and Graduate Teaching:

In addition to its potential use as a source of information for the creation of case simulations, the VMIMS database can be used directly in a variety of ways in the teaching of both undergraduate and graduate students. At the simplest level use of the VMIMS patient file can enhance the experience of the student who is working in the clinical area and is directly involved in patient care. A patient's record can be quickly and efficiently reviewed allowing the student to become familiar with the case, including any previous problems which may have a bearing on the situation with which the student is currently confronted. Student learning as well as quality of patient care are both enhanced in the process. Under the guidance of an instructor students can "work through" a case, either that of a patient currently presented or a case previously selected by the instructor for review. By beginning with basic admissions data and then revealing more data from other areas as students request it the instructor can create the situation whereby the students actually "solve" the case. Reasons for the selection of a particular procedure, contribution of the result to diagnosis and patient management and possible alternative approaches can all be discussed at this time. This approach seems best suited for use with small groups of students in order that maximum interaction and involvement can be encouraged. Alternatively, cases previously selected by the instructor can be used to illustrate points made in more formal lecture, laboratory or seminar settings. The advantage of using the database directly is that a patient's entire file is available at any time and can be used immediately to answer questions which may not have been anticipated when the material was initially selected for presentation. Even in a formal classroom setting this approach allows for a greater degree of interaction between the instructor and the student. The feeling that cases and situations are relevant is enhanced through use of "real-life" rather than simulated cases or text-book examples.

Under the "search and gather" function the data base may be used in an informal setting to illustrate the range of changes which may occur associated with a particular disease syndrome by selecting and examining data from several cases of the disease of interest or alternatively, to investigate the problems associated with various groups of individuals having certain characteristics in common. This information can be used to increase the teaching value of cases seen by the student during clinical duty rotations or as a supplement to and reinforcement of information available in a textbook. Using the database in this way encourages students to view learning as an active process of discovery rather than a passive process of memorization. Once again, the fact that "real" cases are being used as examples seems to appeal to the students and increases their interest in the topic under discussion.

At both graduate and undergraduate levels access to the database can provide material for seminar presentations and papers allowing students the chance to actually seek out and manipulate data rather than relying solely on previously published textbook and journal information. This approach tends to lend interest to the topic and adds emphasis to the hypothesis that learning is an ongoing process requiring action and critical thought on the part of the student.

From our limited experience with use of the VMIMS database for teaching purposes it appears that students accept the concept readily and continue to use the database on their own to investigate and review clinical cases. Learning to use either "show record" or "search and gather" functions requires only a minimal amount of instruction as both functions provide comprehensive menus and on-line help.

There are some drawbacks to using the database directly in teaching not least of which is cost. Purchase of a computer dedicated to the VMIMS database should help to partially alleviate this problem. The speed with which data can be accessed can also be a problem particularly in formal classroom settings or at periods of peak use since the system currently shares time on the university mainframe computer. Again the purchase of a separate computer system, dedicated to the support of VMIMS, would help to solve this problem. Finally, as previously mentioned there are gaps in the record related to certain specialty areas which are not currently on-line. In spite of these disadvantages however, VMIMS can be used in a manner which is both effective and interesting to increase the exposure of students to case material, foster a problem-solving approach to patient management and develop an active involvement on the part of the student in the learning process.

Our initial attempts at using the VMIMS database for teaching have been quite limited in scope although both student and

instructor enthusiasm provide encouragement for further experimentation and development. It remains to be proven however, whether this form of computer-assisted instruction shows any advantages over more traditional forms. Performance and success of students as practitioners must be the ultimate standard.

References

Beck AL, Bergman DV. Using structured medical information to improve student's problem-solving performance. J. Med. Education 61:749-756, 1986.

Bresnitz EA, Stettin GD, Gabrielson IW. A survey of computer literacy among medical students. J Med. Education 61:410-412, 1986.

Diserens D, Schwartz MW, Guenin M, Taylor LA. Measuring the problem-solving ability of students and residents by microcomputer. J. Med. Education 61:461-466, 1986.

Evaluation of Medical Information Science in Medical Education as adopted by the Executive Council of the Association of American Medical Colleges, Washington, DC. J. Med. Education 61:500-543, 1986.

Garrett TJ, Ashford AR. Computer-assisted instruction in patient management for internal medicine residents. J. Med. Education 61:987-989, 1986.

Jensh RP, Veloski JJ. Program for increasing use of computers in medical education. J. Med. Education 61:137-139, 1986.

A COMPUTER SCIENCE SUMMER CAMP

D. A. Swayne[1]

J. VanDuren[2]

M. M. Shepherd[1]

[1]University of Guelph,
Guelph, Ontario
and
[2]Athabasca University,
Fort McMurray, Alberta

ABSTRACT

An overview is presented of the University of Guelph's Summer Computer Camp for the years 1983-1986 inclusive.

We describe main activity areas, breadth and depth of curriculum, and camper and parent responses. The low level of participation of females in each of the four years is discussed. The reactions of veteran (3 or more years) campers to the curriculum and to the camp concept are documented and evaluated.

June 30, 1987

A COMPUTER SCIENCE SUMMER CAMP

D. A. Swayne[1]

J. VanDuren[2]

M. M. Shepherd[1]

[1]University of Guelph,
Guelph, Ontario
and
[2]Athabasca University,
Fort McMurray, Alberta

1. INTRODUCTION

From 1983 until 1986, the University of Guelph Departments of Continuing Education and Part-time Studies and Computing and Information Science jointly operated a very successful Summer Computer Camp. The success of the camp indicates to us that this type of endeavor can act as alternative educational vehicle for the dissemination of computer science instruction.

The decision to hold the camp had been made in 1982. It was noted that a number of camps were being advertised which emphasized group instruction on a single brand of machine in a single language (usually BASIC), and it was obvious that the Guelph camp needed to offer a unique experience in order to succeed. We decided to promote a computer camp that was primarily residential, with virtually unlimited computer access, instruction by university faculty and senior students, and exposure to a wide range of equipment and a variety of languages. Because of these decisions it was possible to attract a camper population which was, for the most part, homogeneous and highly motivated, with both the desire and the ability to take advantage of the situation.

2. MARKETING

The major advertising was accomplished by a mailing to every school board in Ontario for distribution throughout their schools in January of each year.

This mailing made it clear to both the potential campers and their parents that this was a serious undertaking. With the application there was required a letter of recommendation from one of a math, science or data processing teacher indicating that the student had the potential to succeed in this computer-intensive environment. It was also requested that each applicant send a letter describing his or her interests in computers and what specifically he or she expected to gain from the camp. The application asked for the age, sex and experience level of the camper. The returned material presented the planners with a useful profile of the camper population which assisted in the curriculum development process.

The contribution of Continuing Education and Part-time Studies was pivotal in the success of the camp. They designed and commissioned all of the excellent graphics which presented an attractive interface with the outside world: advertising, certificates of accomplishment, camp pictures, certificates of appreciation to donors and guest speakers, and the interface with the University departments providing physical resources.

3. CAMP PROGRAM DEVELOPMENT

The camp program proceeded along two tracks: academic, and recreational and social programs.

3.1. CURRICULUM

Activities were developed for three streams: **rookies, hackers,** and **gurus.** Our experience the first year indicated that, in fact, the hackers (those who fancied themselves as "advanced") should be considered for the most part as campers with a higher level of exposure to computers as opposed to youngsters with a high degree of expertise. The few genuine hackers, with the gurus, were accommodated by special projects, and advanced tutorials. Ability level was the primary discriminant - not age.

A non-systematic search was made to find courseware or software which would have both an intrinsic value from a computer science point of view and appeal for the 12 - 16 year old campers. The computer curriculum for the week was designed to have a balance between classroom and hands-on time. All

campers had a formal introduction period in the first day to all of the tutorials, and were expected to attend at least three formal tutorials each day. Six slots per day were available for formal tutorials, free computing or individual instruction. Other blocks of free computing time were also scheduled.

The following main areas were featured, with rookies and hackers getting different depth of exploration in parallel or offset schedules: an introduction to computing including some history and basic concepts; instruction in BASIC; computer arithmetic and computer architecture; a lecture and hands-on demonstration on artificial intelligence and machine learning; Videotext, including a demonstration of Telidon page creation facilities; compilers, interpreters, stacks, FORTH, and similar language concepts and examples; robotics and the future of the workplace; simple graphics.

The curriculum support included Telidon exposure, LOGO, a programming exerciser called JOSEF, several FORTH machines, a TEACHMOVER robot arm, a hardware lab which included a computer-controlled model car, the resources of the local microcomputer club and the Department's complement of teaching equipment. In the second year, we added a laboratory containing two Apple IIc's, two PCjr's, two Macintoshes, and an Apple Corporation Lisa. Also for the 1984 and 1985 camps, we established a laboratory using BBC Acorn microcomputers. A large number of IBM PCs were used in special-purpose labs (eg. LOGO and Wordstar).

For the third and fourth years, we ran parallel sessions of 6 3/4-hour tutorials each day, and the streams were commingled to allow for differing skill levels. For example, Pascal and BASIC were offered in parallel, as were graphics and LOGO.

In the curriculum of the 1986 camp, the students were given an intensive Prolog tutorial, in order to compare their reactions to it and more traditional language forms.

Prolog is a declarative language in which facts and rules are represented in a database. The data base can be interrogated to give yes or no answers, or in ways in which it will return facts that satisfy given conditions. It requires a very different way of thinking about problems from the procedural approach of most

other common computer languages.

The students were shown some facts about family membership and they became very active at entering facts (or fiction) about their own families and characteristics of some of the members. Their familiarity with the electronic mail system led to them entering facts in one another's directory - not all of this activity was appreciated by the other students!

The concept of rules was introduced and the general schema for recursive rules was given with some suggestions for applications. The intention had been to allow them to build a series of toy expert systems of increasing complexity. We supplied them with a number of basic tools (rules) to enable them for interest, to print out all the elements on a list without them needing to know how lists are represented in Prolog.

As the camp progressed, more of the students abandoned Prolog and switched either to games or to electronic mail. However, a few students became engaged with Prolog and one student undertook not only to help others but to successfully debug some non-working Prolog versions of the mini expert systems.

The Prolog sessions showed that most of the campers enjoy data entry if the directions governing it could be quickly absorbed. They were quite prepared to become very involved in the basically dull story line about family membership. However, the majority of them were much less interested in thinking about a situation in the different framework required by logic programming. The younger campers had considerable trouble with the spelling required to interrogate the data base and that was a source of real frustration for them. It should be pointed out that the one or two students that became totally absorbed in the process made incredible progress and advanced further than we could have anticipated.

Every camper received at registration a binder containing prepared notes and background information on most areas of the curriculum. The classroom instruction topics were presented in such a way that campers were required to make a few notes in appropriate windows on the sheets. The remaining material contained examples, basic vocabulary and syntax for the different systems and historical and theoretical material for those who developed a particular interest in an area.

The staff was responsible for carefully monitoring the use of scarce resources such as the robot arm and encouraging interest on the part of those few campers who weren't self-motivated. After the introductory instruction in each area the tutors became primarily resource people and thus had the opportunity to observe the reactions, abilities and problem-solving approaches of the campers as well as to judge the relative value of the various activities.

3.2. RECREATIONAL AND SOCIAL PROGRAM

The Continuing Education and academic staff cooperated in the development of recreational and social activities, but the Division was responsible for all of the non-academic issues. The success of the camp is in large part a reflection of the careful planning and execution of the "after-hours" activities.

The day was organized from 7:00 am rise-and-shine to 9:30 late snack and bedtime. There were approximately five hours per day of scheduled computer activity (30 hours total during the week). Sports and outdoor games were interspersed throughout the camp day to release frustrations and ensure physical well-being. Evening programs involved such events as Japanese language and writing, amateur astronomy, a video games night, a lecture on George Orwell and 1984 (in 1984), theatersports (an interactive and competitive form of skit-theater). and current movies with a high content of computer graphics. Two outings were planned each week: a get-acquainted barbeque and a combined wrap-up party and awards night.

4. THE CAMPER POPULATION

The camp was held in three one-week sessions in the first two years and four such sessions in the latter two years. Limits were set on the number of campers accepted. In this way we could ensure the high degree of individual attention as well as maintain the one-to-one ratio of camper to machine. Both of these were factors which we felt were necessary for a quality experience.

In all years we drew campers from a broad geographical base. We had a considerable number from Northern Ontario and from rural and small town environments, as well as from major metropolitan areas. Since many of our campers are

in elementary schools and from non-urban areas, it would seem that we attracted youngsters whose desire for computing knowledge is not being met in their schools or communities. We attracted children with physical limitations as well, and they felt comfortable enough with our environment that they returned for subsequent camp years.

Our percentage of female campers stayed constant over four years at less than 20%. Their skills and interest varied widely, with only one or two who showed interest or aptitude. It would seem that the selection process has been at work on females in our school system (at least among those likely to apply to our camp).

The age percentages are as follows. (Again, they have remained relatively constant for four years.)

age	12	13	14	15	16
percentage	<30	30	20	10	10

The registration varied from 104 camper weeks in 1983 to 160 camper weeks in 1985.

5. AWARDS PROGRAM

Our camp was expensive and did not make a profit. The result to prospective campers was a steep fee which we hoped to defer through a fellowship or scholarship program. The money was to be obtained from corporate donors.

In the first year of operation, the number of scholarships was fairly large - 54 - but most came from the College of Physical Sciences at Guelph.

In the second year, the number of scholarships decreased, to 39, but they represented bona fide donations from corporate sponsors Canada-wide.

In our final year, we were able to offer some bursaries and full scholarships. In a presentation ceremony the campers got to meet representatives of the company donating their scholarship. The Department faculty and University personnel were able to meet with company executives.

One company executive, was an eighteen year-old president of a software firm, who donated two scholarships. Her letter to us indicated wistfully that she

would have liked such a camp when she was young.

6. POST-CAMP ANALYSIS

We received feedback on the camp from three sources: the campers, their parents and the staff. Longstanding campers were also queried in order to find what made them return each year for several years.

6.1. EVALUATION AND FOLLOW-UP

A short questionnaire was given to the campers at the end of the week to help us judge their reaction to the curriculum. We were interested in their perception of the total time they had spent at the computer, the time spent at each activity, and how much they felt they had learned about the various subjects.

The average total time per week spent on computing activities was claimed to be in excess of 34 hours, although the total time actually scheduled for these activities was usually 30 hours. These additional hours could be actual as opposed to perceived since a number of the campers were persistent in their desire for extra computing time - missing snacks, taking short meal breaks etc.

Each student was asked to estimate the time spent on one area which was his or her primary or most intensive activity. The average response was 12.0 hours. The second or next area of involvement averaged 8.0 hours of camper's time. As indicated by these figures the average camper was introduced to a wide variety of topics and from these chose one or two to explore in some depth.

The **parents** of the campers were sent a questionnaire several weeks after the end of the camp session. The major part of the information trapped in this process regarded the camp in general as opposed to the computer curriculum specifically. There was a 45% return rate on this evaluation summary. There was a unanimous expression of satisfaction with the camp from these respondents and 84% would encourage their children to return another year. All of the parents felt that their campers had increased their computer knowledge at the camp and there were many additional positive remarks indicating satisfaction with the curriculum, the staff and the general level of organization.

6.2. LONGITUDINAL PERSPECTIVE

A veteran of three camp years (1984-1986), LaMaia C. agreed to an interview in order to allow us to present the camp from a consumer perspective. The length of her involvement with the camp means that she saw it in various stages of evolution, and from the point of view of most of our target chronological ages.

The following are the questions asked, with her answers.

1. Would you attend the camp again? Could you explain why or why not? *Certainly. I enjoyed the camp a great deal and it seemed a worthwhile thing to do.*
2. How did you feel about the overall camp program? *It's a good way to run a program. Having a choice of classes is better than having them compulsory.*
3. If you were running the camp which of the existing parts of the program would you leave out? *I'd stop making the sports mandatory.*
4. If you were running the camp what additional programs would you add to the existing program? *I'd have more hardware and building - like Howie (the hardware tutor) was doing.*
5. How did you feel about the Prolog tutorial? *It was fun to learn something that I hadn't considered playing with before.*
6. What do you remember as the differences between a declarative language and the imperative programming languages you had used before? *Don't know. (the question was explained) Prolog seems to be perpetually recursive.*
7. What did you learn at the Prolog tutorial? *I learned basic Prolog and have retained most of it. I'll be using Prolog during the last few weeks of school in the computer science class as the last topic in the course.*
8. What did you learn at the C tutorials? *We learned the basic rules about C. I knew structured programming techniques before otherwise I would have learned about structured programming too.*
9. What did you learn at the graphics tutorials? *I didn't do this tutorial. Most of the people didn't know enough C to to interesting graphics so they taught C. The queues for the IRIS were too long so I didn't use them either.*

10. What can you recall about facts and rules in Prolog? *I remember that data structures could be easily accessed.*

11. What can you recall about the Unix operating system? *I've been using VMS and UNIX was annoying since there were no help files and you felt that the designers had made it as difficult as possible to use the system.*

12. What can you recall about text editors? *Text editors are fun but they are not good on UNIX the Turbo text editor is superior to anything I've used on a mainframe.*

13. What can you recall about looping and transfer of control in C? *I'm not really fluent in C. Structurally C is about the same as Pascal. Serious errors in the control structures are more possible than with Pascal.*

14. What can you recall about file pointers in C? *Very little.*

15. What can you recall about recursion in Prolog? *Recursion is almost permanent in Prolog. Prolog likes backtracking and behaves as it everything is on a stack. Prolog is happy to pop stuff off the stack.*

16. Have you used anything of the things you learned at camp since going home? If so, how much? *Prolog will be used during the last few weeks of school. I used C for about a month in the fall and then went back to Pascal. I've used lots of things that I learned on my own - like sets in Pascal - during the free time. I've been playing with the robot here so I used what I learned using the robot arm.*

7. CONCLUSIONS

We are obviously proud of the camp. It has been largely self-supporting, has shown our University and Department in a positive light, and has not caused an unusual burden on the resources of the Department. Manpower requirements from the academic side are roughly equivalent to one single semester course for the academic coordinator, several teaching assistants, and a few hours of volunteer instruction from other faculty. The key ingredient in the success of this venture has been the quality of the marketing effort and the overall organization by Continuing Education.

The camp has given us the opportunity to interact directly with bright students in the middle years of secondary school. A mainstream view of computing, tailored to their skill and age levels, has been presented. A small Department with a limited liaison budget has been able to indulge in a low-key recruiting effort, and to advance the state of knowledge (significantly) of the individuals who came to the camp.

A Prototype Expert System For Teaching Brain/Behaviour Relationships

Mark Szwarc*, Subir Bandyopadhyay, George Lasker,
and David Reynolds**
Working Group in Cognitive Studies
University of Windsor
Windsor, Ontario

Introduction

A number of major neuroanatomical systems can be identified in the brain that mediate different but overlapping functions. This software is a prototype of a rule-based (expert) system, written in prolog, that can be used to help students learn brain/behaviour relationships (see, for example, Fig. 1). A student using this program may describe behavioural functions and, in reply, the program suggests possible causes for this behaviour in terms of neuroanatomical activation or inhibition. Similarly, the student may describe neuroanatomical activation or inhibition and the program indicates what behaviour may result. The rule-based nature of the system allows the user to ask how particular results are generated and how other results may be inferred.

Some Sample Rules

The rules that make up the core of this software have progressed through a series of modifications. Some examples of the final rules are:

---

* Questions regarding the microProlog programming may be addressed to the first author, Department of Computer Science, University of Western Ontario.

** Questions related to brain/behaviour relationships may be addressed to last author, Department of Psychology, University of Windsor.

cerebral-cortex

precise-motor-functions

precise-sensory-functions

complex-cognition

memory-trace-storage

limbic

reward

processing-into-permanent-memory

attack

hypothalamus

reflex-emotion

homeostasis

gender-specific-behaviour

gender-differentiation-growth

reticular-formation

consciousness

muscle-relaxation

Figure 1. Major brain/behaviour relationships.

R5 Activation of a system implies that the components of that system are activated to the same degree.

R1 Activation of a structure which is part of a system generaly implies that the system is also activated but to a lesser degree than the specific structure.

R3 Activation of a structure generally increases the function associated with that structure.

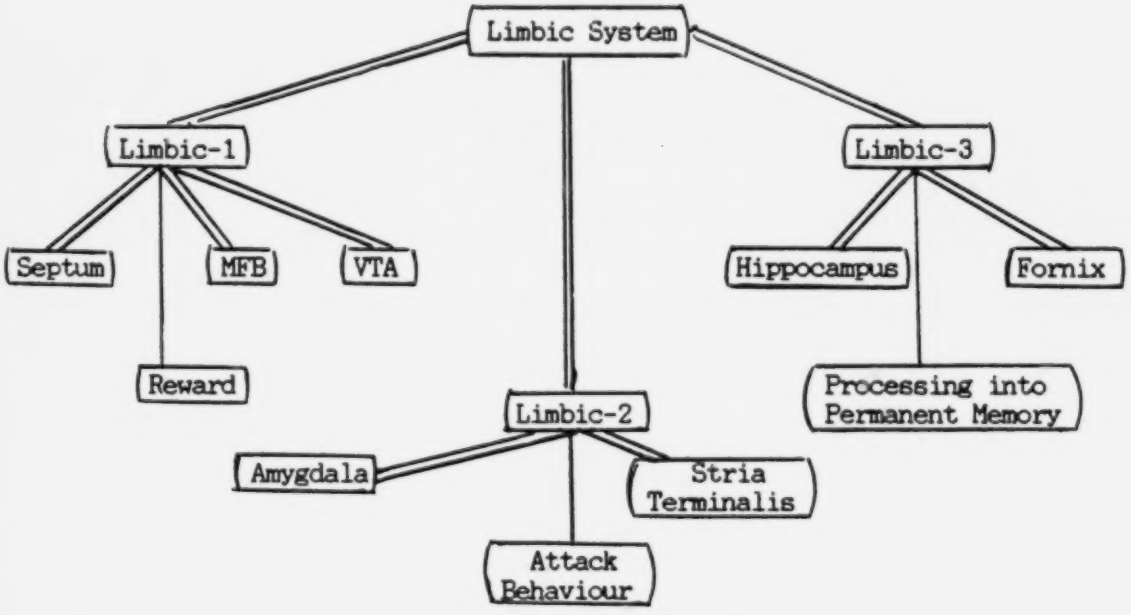
A model exists which defines the manner in which the rules are stored and manipulated. This means that there is a standard format for the rules and an algorithm (inference engine) for applying the rules to draw conclusions.

Representation of Knowledge

In this system, the domain knowledge is one of three distinct types. First, there is a set of facts that define the neuroanatomical systems, structures, functions, and the relationships among these entities. This level is represented in the form of a semantic network (see Fig. 2). The semantic net defines the systems and structures of the brain in the form of a hierarchy. In prolog, this network is made up of predicates. The predicate called **part-of** relates a structure to the system in which it belongs. For instance, **part-of (septum, Limbic-1)** indicates that the septum is one component of a sub-system labeled Limbic-1.

Another very important relation is named **associated-with**. This relation stores the association between brain systems and functions. For example, one subsystem within the limbic system has been shown to mediate positive reinforcement (reward). A fact is stored representing this information and

this fact is associated-with (Limbic-1, reward). Arbitrary labels, such as



Key: P = Part-of ═══
A = Associated-with _____

Figure 2. Partial Semantic Network.

Limbic 1, have been assigned to some brain subsystems that mediate functions because no formal name exists for these neuroanatomic subsystems.

The next level of knowledge is a set of transient or temporary facts. This set is made up of facts supplied by the user and facts that are inferred by the system. These facts define a particular state of the system or structure being modeled. For instance, if the user enters the fact that there is activation in the septum, this would be stored as a temporary fact. Also, if the system were to evaluate a particular situation, the conclusions drawn would also be stored as temporary facts. These can be contrasted with the facts that make up the semantic network (described above) that defines the structures and relationships within the brain that are never changed. Once the user decides that he/she wishes to explore a different situation, temporary facts must be deleted before new facts are entered.

The expert's rules constitute the third level of domain knowledge. They define how temporary facts and permanent facts may be used to draw conclusions. The rules are encoded using the following model:

<primary conditions> and <secondary conditions> imply <conclusion>

If the primary conditions and secondary conditions are proven to be true, then the conclusion is asserted to be true and will be stored as a temporary fact. The motivation behind the separation of conditions into primary and secondary

sets becomes apparent when the algorithm of inference is examined.

The Inference Engine

The inference engine is the part of a rule-based system which directs what rules will be used and how they will be applied. In this program rules are stored in the order in which they are to be applied. This method of scheduling the rules is sufficient because the scope of the system is limited. When processing is initiated, the inference engine will select the first rule, evaluate it, then select the second rule, and so on. When a rule is processed, the primary conditions are tested first. If there is a primary condition that is proven false, then the rule is abandoned. If all of the primary conditions are true then the rule succeeds and the conclusion is stored as a fact.

The intricacy of the model is revealed when there is a case where the conditions of a rule cannot be shown to be either true or false: this is termed an "unknown condition." First, consider the situation where all primary conditions are true and the secondary conditions are either true or unknown. In this case the user would be asked to assign a truth value to the unknown conditions. If a situation exists where one may expect the user to be able to resolve an unknown truth value, then this condition is included in the secondary list.

When the primary list contains only true and unresolvable conditions, then the secondary list is checked. If any secondary condition is false the rule is abandoned. Otherwise, it is desirable to try this rule again after more facts have been inferred. It would be a waste to evaluate the true conditions again so a new temporary rule is created that consists of all those conditions from the original rules that were unresolved. Of course, the

conclusion stays the same in the new rule.

As indicated above, each rule is checked in sequence. After all original rules have been evaluated, the set of temporary rules is processed in an identical manner. When there is no new fact added to the fact-base as a result of a successful rule, the process terminates.

The inference engine also has the ability to use defined rules in a backward manner, working from the conclusion to the conditions. That is, given a fact, the conditions required for a rule to assert this fact are discovered. This process is obviously not the same as that described above. In this case of reverse reasoning, a rule is found that has a conclusion that matches the given fact. Then an attempt is made to prove each condition of that rule. Some conditions correspond to permanent facts and may easily be proven to be true or false. The conditions that correspond to temporary facts are displayed and identified as required to satisfy the rule. Example number 3, below, outlines the approach.

Examples

The examples in this section do not demonstrate the difference between primary and secondary rules, but simply show how facts and rules are used to assert conclusions.

Example 1: The user wishes to know the result of activation of the septum (the facts and rules referred to in this example are stated above).

- the temporary fact that activation of the septum exists is stored
- rule R1 is applied:

R1 Activation of a structure which is part of a system generally implies that the system is also activated but to a lesser degree than the specific structure.

septum is activated and

septum is part of Limbic-1

THUS store temporary fact that activation exists in Limbic-1

- rule R3 is applied:

R3 Activation of a structure generally increases the function associated with that structure.

Limbic-1 is activated and

Limbic-1 is associated with reward

THUS store the temporary fact that increase in reward exists

Example 2: The user wishes to know what the possible outcomes of activation of the limbic system will be (note that rule 5 occurs before the others).

- the temporary fact that activation exists in limbic system is stored

- rule R5 is applied:

R5 Activation of a system implies that the components of that system are activated to the same degree.

limbic system is activated and

Limbic-1 is part of limbic system

THUS there is activation in Limbic-1

Limbic-2 is part of limbic system

THUS there is activation in Limbic-2

Limbic-3 is part of limbic system

THUS there is activation in Limbic-3

- rule R3 is applied:

R3 Activation of a structure generally increases the function associated with that structure.

Limbic-1 is activated and

Limbic-1 is associated with reward

THUS there is an increase in reward

Limbic-2 is activated and

Limbic-2 is associated with attack

THUS there is an increase in attack

Limbic-3 is activated and

Limbic-3 is associated with processing into memory

THUS there is an increase in processing into permanent memory

Example 3: The user wishes to know what might cause an increase in consciousness.

- R3 is selected as a possible rule because the conclusion of the rule fits the fact that is being investigated:
- R3 Activation of a structure generally increases the function associated with that structure.
- This rule indicates that if an increase in some function is observed then the structure that is related to that function is probably activated.
- The permanent fact Reticular-Formation-1 is associated with consciousness is located.
- Conclude that activation of Reticular-Formation-1 would cause an increased level of consciousness.

(Note. Reticular-Formation-1 is a label assigned to a subsystem of the reticlar formation.)

Future Modifications

The first major modification that should be attempted is the streamlining of the inference engine. This, coupled with a faster computer and a hard disk drive, would increase the speed of execution of the system to an acceptable

level.

Another major addition would be a natural language interface. As it stands, the user is presented with labels and short forms for many terms, and the user is forced to interact with the system using keywords. A natural language interface would make the system easier to learn, easier to use, and easier to understand.

General Bibliography

Although many sources were used in the preparation of this manuscript, the following are appropriate as major references.

micro-PROLOG and Expert Systems

Clark, K. L., & McKabe, F. G. (1984). micro-PROLOG: Programming in logic. New Jersey: Prentice-Hall.

Hayes-Roth, F., Waterman, D. A., & Lenat, D. B. (1983). Building expert systems. Reading, MA: Addison-Wesley Publishing Co. Inc.

Brain and Behavior

Angevine, J. B., Jr., & Cotman, C. W. (1981). Principles of neuroanatomy. New York: Oxford University Press.

Carlson, N. R. (1985). Physiology of behavior (3rd ed.). Toronto, Ontario: Allyn and Bacon, Inc.

Shepherd, G. M. (1983). Neurobiology. New York: Oxford University Press.